Voice over IP Service Description

Columbia University Information Technology

April 24, 2007
Revision : 1.16

Contents

1 Introduction

Columbia University Information Technology (CUIT) has embarked on a project to implement Voice over IP (VoIP) service initially as a replacement legacy PBX technology and further as first step in the strategic direction of converged network services. Motivations for this change include:

- Obsolescence of the legacy Rolm PBX: The PBX was declared end-of-life in 1998 and end of support is expected within the next three years. As of October, 2008, when our current three year maintenance contract expires, Siemens has already stated that they will only offer one year renewal contracts with a commitment to give customers a one year notice of end of service. We've already seen Siemens' difficulty in keeping the current system running due to the age of the technology. The scope of the Rolm PBX installation at Columbia covers 12 sites and about 20,000 phones.
- Opportunity for convergence: The need to replace the PBX affords us the opportunity to converge the voice and data networks. Convergence has happened organizationally as of March, 2006, with the voice, data and video groups having been combined. It is also the right time for convergence: VoIP technology is mature as is the underlying IP networking.

After much research and analysis, CUIT has established an architecture and implementation that is described in this document. We are cooperating with our peers at the University of Pennsylvania, MIT and the University of North Carolina who have all made like technology choices (as have many others). This collaboration includes knowledge and code sharing for which we are grateful for our colleagues' help.

The initial Phase I implementation of the service will involve approximately 700 phones in the Studebaker building at 615 West 131st Street in the Manhattanville neighborhood of Northern Manhattan. Implementation commenced in February 2007 and runs through October as departments move into Studebaker. The focus of Phase I is basic PBX replacement telephony service.

Phase II development, expected to begin in mid-2007, will address advanced capabilities offered by the use of VoIP for real time communications.

This document describes the overall CUIT VoIP architecture as well as specific information regarding the Studebaker implementation which will serve as a model for ongoing VoIP implementation campus-wide.

For those customers currently on a CUIT-owned phone exchange (212-851,853 or 854), their existing numbers will not have to change as they will be moved from the current PBXes they are on to the new VoIP system.

2 Architecture

The VoIP system architecture is outlined in figure 1. Key principals of the architecture are **resiliency** from single points of failure implemented by redundancy and independence of components; **simplicity** such that the system is easily understood and maintained; leveraging the use of widely accepted and available **open standards**, namely the Session Initiation Protocol (SIP)[1], IP, Ethernet, and Unix standards; **cost effectiveness**; and **scalability**, as this system is expected to grow from an initial implementation of 700 lines to eventually replace 20,000 traditional PBX lines and accomodate growth of the new Manhattanville campus.

The following sections describe each component of the architecture in detail.

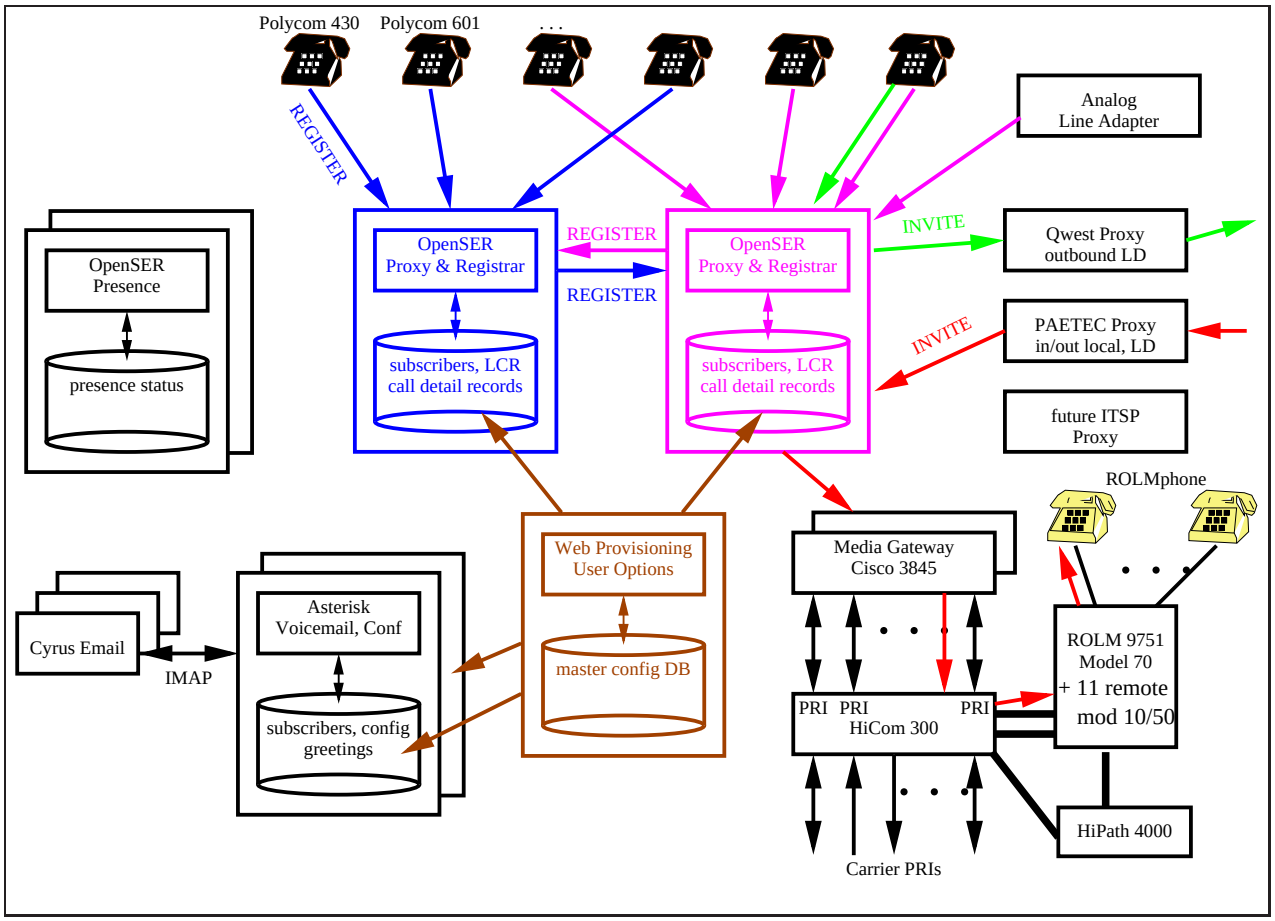

Figure 1: VoIP Architecture and PBX/PSTN interconnections. Red indicates a sample inbound SIP-to-Rolm call. Green indicates a sample outbound VoIP-to-PSTN call

3 User Agents (phones)

For the initial implementation phase, we have chosen to use "hard phones" for our SIP User Agents (UA). Later support of "soft phones" – software that runs on a desktop PC – as well as WiFi phones is anticipated but is explicitly excluded from the initial project scope. At this point, the goal is basic PBX functionality at a $200 per phone average price point.

The hard phones connect to Ethernet switch ports that provide IEEE 802.3af power over Ethernet (PoE). IEEE 802.1q VLAN tagging is used, along with Cisco Discovery Protocol (CDP), to dynamically select the appropriate voice-only VLAN to isolate VoIP from other data traffic. Isolation enables guaranteed quality of service (QoS) and enhanced security. See section 14.8 on page 43 for more information.

While the use of the phone's built-in Ethernet switch (if any) is supported where available network ports are constrained, our preference is to not use the phone for anything but VoIP. This eliminates an unnecessary interdependence between the phone and computer. For example, current affordable IP phones have a 100 Mbps fast Ethernet data port while many desktop and laptop computers today have a gigabit Ethernet port. In the event that the phone's switch is used however, the native VLAN is used for the data port and voice traffic is trunked on the voice VLAN, isolating voice and data traffic.

3.1 Phone Models and their Features

After a peer review and testing of products from Cisco, Polycom, Snom, and Siemens, we have selected the Polycom SoundPoint IP 430 and 601 as the main models. The Polycom phones are cost-effective, physically robust and attractive, implement all required and many optional SIP features, have good sound quality, and are from a recognized force in the voice and video conferencing industry. Phones are purchased from one of several competitive distributors and technical support is provided directly by Polycom through their Premier customer support program. We are considered tier 3 support customers of Polycom, meaning we are a "service provider" providing direct tier 1 and 2 support to our end users. Several CUIT staff have completed certification exams required by Polycom to qualify for direct support.

All Polycom phone models share the same basic feature sets. The Polycom 430 is a 2-line handset with full-duplex speaker phone, RJ11 headset jack, medium-sized monochrome LCD display, and RJ45 Ethernet data jack. The 430 will replace the functionality of the current RolmPhone 120 basic model, a single-line phone with no display or speaker phone, as well as many current RolmPhone 624s which have 6 lines, a display and speakerphone.

The Polycom 601 is a 6-line phone, expandable to up to 48 line appearances supporting up to 12 simultaneous calls with "sidecars". It has a somewhat larger monochrome display and is otherwise identical to the 430. It is based on an older generation of hardware however so boots more slowly and is not as future-proof as the 430.

Other models from Polycom that will be used in limited numbers are the SoundPoint IP 4000 which is a conference room phone and possibly the SoundPoint IP 550 and 650, new high-end models with a back-lit display and addition of a 7 kHz wide-band audio G.722 codec to supplement the usual G.711 and G.729 narrow-band codecs. We are also investigating the new 320 and 330 low-end models. The 320 has not built-in Ethernet switch.

3.2 Phone Provisioning

The goal of phone provisioning is "zeroconf". That is, nothing need be done to provision a phone received from the factory other than plug it in to the "provisioning subnet". Furthermore, the phone must be provisioned securely such that a packet sniffer can not discover the SIP user's password nor can one obtain this information using a forged

provisioning request. This is accomplished through AES 128-bit CBC encryption of the configuration file. The initial AES shared encryption key is downloaded in the clear to the phone when first bootstrapped on the secure provisioning subnet.

3.2.1 Polycom Phone Provisioning Details

Polycom phones support provisioning via TFTP, FTP and HTTP/HTTPS protocols. The provisioning server (boot server) is learned via DHCP option 66 (TFTP server) and is provided as a URL. The HTTPS server is configured to only permit access from the secure provisioning subnet. The initial bootstrap of the phone downloads the latest BootRom and SIP application images as well as the initial encryption key. The URL for the provisioning server is written to the phone's "device" flash memory such that further boots, even where DHCP service does not provide the provisioning server URL, are pointed at the "runtime" provisioning service. Files provided by this service are encrypted using the aforementioned AES key. This can include the file containing a new encryption key. Keys will be changed periodically such that the AES key will not be compromised over time. Each phone is provisioned with a unique AES key, HTTPS user and password.

Phone provisioning files consist of:

MAC-specific configuration A file named by the phone's MAC address, *MAC*.cfg, is requested. If this file is not found, the phone requests a generic configuration file, 000000000000.cfg. The MAC-specific configuration contains the name of the SIP application image file and a list of other configuration XML files to load. XML tags in these files override defaults from files listed later in the list. Polycom-provided default values are in this way overridden by CUIT-provided values such as local preferences like ring tones, dial plan and SIP user registration information. See figure 2 for an example.

Generic configuration The generic configuration (figure 3), named 000000000000.cfg, is set up such that an unprovisioned phone boots with the current version of BootRom, SIP application, device flash setting for the provisioning server, and a campus-only restricted phone extension (x77999). This allows for testing as well as discovery of "unprovisioned" phones on the network: Their registrations show up on the SIP proxy and they are able to place on-campus restricted calls.

SIP registrations SIP user and server settings are in a per-phone registration file, *MAC*-reg.enc, figure 4. This file is encrypted with the current AES key.

Factory default phone configuration Factory default phone configuration settings are found in phone1.cfg. This file is never modified other than to replace it with newer factory default configurations that are provided with each new Polycom firmware release.

Factory default SIP settings Factory default SIP settings are found in sip.cfg. This file is never modified other than to replace it with newer factory default settings that are provided with each new Polycom firmware release.

local overrides CUIT's local overrides of the factory default sip.cfg and phone1.cfg are set in the local-settings.cfg file, figure 5.

Initial contact directory A default initial local contact directory (speed dials), named 00000000000-directory.xml is downloaded by the phone at initial boot, if available, and is used to seed the local phone directory with initial values. See figure 7 for a sample directory file. Phone directory entries can be edited by the phone user directly, for example, by saving the caller ID information for a received or placed call. Also, the local directory can be synchronized and updated centrally.

The phone specific configuration files are generated from templates by the Web-based phone provisioning application, described below. The template files are named MAC.cfg and MAC-reg.cfg (literally "MAC").

```
<?xml version="1.0" standalone="yes"?>
<!-- Columbia Polycom phone master config for MAC addr 0004f204ef3a -->
<APPLICATION APP_FILE_PATH="sip.ld"
  CONFIG_FILES="0004f204ef3a-reg.enc, phone1.cfg, local-settings.cfg, sip.cfg"
  MISC_FILES=""
  LOG_FILE_DIRECTORY=""
  OVERRIDES_DIRECTORY=""
  CONTACTS_DIRECTORY=""/>
```

Figure 2: Sample Polycom *MAC*`.cfg` File

```
<?xml version="1.0" standalone="yes"?>
<!-- Default Master SIP Configuration File-->
<!-- Edit and rename this file to <Ethernet-address>.cfg for each phone.-->
<!-- $Revision: 1.16 $  $Date: 2007/04/25 02:30:18 $ -->
<!-- customized for Columbia defaults -->
<APPLICATION APP_FILE_PATH="sip.ld"
 CONFIG_FILES="default-reg.cfg, phone1.cfg, local-settings.cfg, sip.cfg"
 MISC_FILES=""
 LOG_FILE_DIRECTORY=""
 OVERRIDES_DIRECTORY=""
 CONTACTS_DIRECTORY=""/>
```

Figure 3: Polycom 000000000000.cfg file, updated to include a default registration.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Columbia overrides for phone1.cfg -->
<phone1>
 <reg reg.1.displayName="Alan Crosswell"
      reg.1.address="10508"
      reg.1.label="10508"
      reg.1.type="private"
      reg.1.auth.userId="10508"
      reg.1.auth.password="10508"
      reg.1.lineKeys="1" />
 <reg reg.2.displayName="Alan Crosswell"
      reg.2.address="10501"
      reg.2.label="10501"
      reg.2.type="shared"
      reg.2.thirdPartyName="10501"
      reg.2.auth.userId="10501"
      reg.2.auth.password="telefon"
      reg.2.lineKeys="1" />
</phone1>
```

Figure 4: Sample Polycom *MAC*`-reg.enc` file (after decryption)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Columbia overrides for sip.cfg -->
<sip>
   <voIpProt>
      <server voIpProt.server.1.address="siptest.columbia.edu"
              voIpProt.server.1.port=""
              voIpProt.server.1.transport="DNSnaptr"
              voIpProt.server.1.expires="3600"
              voIpProt.server.1.register="1"/>
      <SIP>
  <outboundProxy voIpProt.SIP.outboundProxy.address="siptest.columbia.edu"
                        voIpProt.SIP.outboundProxy.port=""
                        voIpProt.SIP.outboundProxy.transport="DNSnaptr"/>
          <alertInfo voIpProt.SIP.alertInfo.1.value="AUTO_ANSWER"
                     voIpProt.SIP.alertInfo.1.class="3"/>
          <alertInfo voIpProt.SIP.alertInfo.2.value="RING_ANSWER"
                     voIpProt.SIP.alertInfo.2.class="4"/>
          <alertInfo voIpProt.SIP.alertInfo.3.value="INTERNAL"
                     voIpProt.SIP.alertInfo.3.class="5"/>
          <alertInfo voIpProt.SIP.alertInfo.4.value="EXTERNAL"
                     voIpProt.SIP.alertInfo.4.class="6"/>
      </SIP>
   </voIpProt>
   <dialplan dialplan.impossibleMatchHandling="0" dialplan.removeEndOfDial="1">
      <digitmap dialplan.digitmap=
        "51,0|51,[2456]xxxx|80,xxxx|9[245],xxxx|72,xxxx|97,xxxxxxxxxx.T|53,xxxx
         |981|[1346]xxxx|7[134567890]xxx|93,1xxxxxxxxxx|93,011xxx.T|911
         |93,[349]11|99|*86|*82,93,1xxxxxxxxxx|*82,93,011xxx.T
         |*67,93,1xxxxxxxxxx|*67,93,011xxx.T|0xxxx.T"
                dialplan.digitmap.timeOut="3"/>
   </dialplan>
   <TCP_IP>
      <SNTP tcpIpApp.sntp.address="sundial.cc.columbia.edu"
            tcpIpApp.sntp.gmtOffset="-18000"/>
   </TCP_IP>
   <HTTPD httpd.enabled="0"/>
   <voice>
      <volume voice.volume.persist.handset="1"
              voice.volume.persist.headset="1"
              voice.volume.persist.handsfree="1" />
   </voice>
</sip>
<!-- Force settings into the phone so it will still phone home even
     when it's provisioning server is not supplied by DHCP -->
<device device.set="1"
        device.prov.serverName.set="1"
        device.prov.serverName="http://www.columbia.edu/~alan/poly/initial"
        device.prov.serverType.set="1"
        device.prov.serverType="3"/>
```

Figure 5: Polycom `local-settings.cfg` File

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Columbia overrides for phone1.cfg -->
<!-- unprovisioned phones will register with extension 77999 -->
<phone1>
 <reg reg.1.displayName="NOT PROVISIONED"
      reg.1.address="77999"
      reg.1.label="77999"
      reg.1.type="private"
      reg.1.thirdPartyName=""
      reg.1.auth.userId="77999"
      reg.1.auth.password="password" />
</phone1>
```

Figure 6: Polycom `default-reg.cfg` file

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<directory>
  <item_list>
     <item>
       <ln>Crosswell</ln>  <!-- last name -->
       <fn>Alan</fn>       <!-- first name -->
       <ct>43754</ct>      <!-- contact URL -->
       <sd>1</sd>          <!-- speed dial index -->
       <rt>1</rt>          <!-- distinctive ring type -->
       <ad>0</ad>          <!-- (don't) divert calls from this caller -->
       <dc />              <!-- automatically divert call here -->
       <ar>0</ar>          <!-- (don't) reject calls from this caller -->
       <bw>0</bw>          <!-- (don't) add to buddy watch list -->
       <bb>0</bb>          <!-- (don't) block from contact's buddy list -->
     </item>
     . . .
  </item_list>
</directory>
```

Figure 7: Polycom *MAC*`-directory.xml` file

3.2.2 Specific Polycom XML tags set by CUIT

The following XML tags are configured by CUIT either as system-wide defaults or on a per-phone and per-SIP registration basis. See figures 2 and 4 for examples. See the Polycom Administrator's Guide[2] for detailed documentation.

voIpProt.server.1.address SIP proxy name to use for DNS NAPTR lookup: sipdev, siptest or sip.columbia.edu, for development, test and production systems, respectively.

voIpProt.server.1.port Must be blank or it overrides DNS NAPTR/SRV values.

voIpProt.server.1.transport Set to DNSnaptr to use DNS NAPTR and SRV lookups to discover the names of the replicated proxies, transport protocol (UDP, TCP or TCP/TLS) and port to use.

voIpProt.server.1.expires The lifetime in seconds for the SIP REGISTER. Set to 1 hour.

voIpProt.server.1.register Set to 1 to cause the phone to send SIP REGISTER.

outboundProxy Settings for outbound SIP proxy. Same as inbound proxy, above.

voIpProt.SIP.alertInfo.x.value,class The value is matched against the SIP Alert-Info header and if matched, the given class is used. Values and classes follow and are defaults supplied by Polycom in sip.cfg.

value	class	description
AUTO_ANSWER	3	auto answer immediately on incoming call
RING_ANSWER	4	auto answer on incoming call after first ringing for 2 seconds
INTERNAL	5	internal call distinctive ring
EXTERNAL	6	external call distinctive ring

dialplan.digitmap Set the phone's dialplan. For example, after dialing a 1,3,4 or 7 followed by four digits, the phone automatically dials. After dialing 93, getting a secondary dialtone, and dialing 011 followed by 3 or more digits, the phone waits 3 seconds after the last digit entered and then dials. The timeout is needed for international calling where the numbers are of varying length.

dialplan.digitmap.timeOut Sets the time to 3 seconds to wait before dialing the call.

tcpIpApp.sntp.address This tag sets the simple NTP time server. We use sundial.cc.columbia.edu as our NTP server.

tcpIpApp.sntp.gmtOffset We are in US Eastern time which is 5 hours West of Greenwich. The value is in seconds East of Greenwich, so we use -18000. See the manual for further SNTP settings, including options to enable or disable overriding the GMT offset and SNTP server address via DHCP.

tcpIpApp.sntp.daylightSavings... We currently use the vendor defaults for the DST settings. Note that since the settings are always relative to the current year, we were unable to install the 2007 DST changes until after January 2007.

httpd.enabled Set OFF(0) to disable the phone's internal web server.

voice.volume.persist.handset,headset,handsfree Set ON(1) so that user-selected volume settings are retained across calls. The default OFF(0) is required by some regulatory agencies to return e.g. a public use phone to nominal volume levels after each use. For a personal desk phone, we want to retain the user's preferences.

device.set Set ON(1) forces device settings to be written to the phone's flash memory as if they had been set from the phone's settings menu. For each setting, device.prov.*setting*.set must also be set ON(1) to make that setting persistent.

device.prov.serverName Sets the provisioning server URL. This is used when a DHCP server does not provide the URL (TFTP server – option 66). So, after provisioning, the phone can be moved anywhere and still knows how to reach its provisioning server. This setting is overridden by the DHCP-supplied TFTP server.

device.prov.serverType Type 3 is used for an HTTP server; 4 for HTTPS.

reg.x.displayname is the display (personal) name of the subscriber.

reg.x.address is the extension number (SIP username).

reg.x.label is what is displayed on the phone LCD panel next to the line key.

reg.x.auth.userid is the SIP subscriber name (same as the extension).

reg.x.auth.password is the SIP subscriber password.

reg.x.linekeys Is the number of times the same extension appears on a line key on the phone. Set greater than 1 stacks incoming waiting calls on successive line keys, rather than having call waiting displayed on a single line. This is analogous to hunt groups in a key system.

reg.x.type Set to private or shared. Shared is used to implement bridged line appearance.

reg.x.thirdPartyName This is the AOR for bridged lines. Usually the same as the reg.x.address.

prov.polling.enabled is enabled(1) to cause the phone to periodically check the provisioning server for configuration and software updates. The default interval of checking daily at 3am is currently used. We plan to randomize this value on a per-phone basis as the system scales up to distribute the load across the provisioning web servers.

3.2.3 Files uploaded by the Polycom phone

The Polycom phone uploads several files to the provisioning server at boot time periodically when the four arrow keys are pressed simultaneously for three seconds:

Boot loader event log *MAC*-boot.log. This file is used for debugging.

SIP application event log *MAC*app.log is uploaded periodically (48 hours default) or when the log file exceeds a particular size (16K default).

Local contact directory Contacts that are updated by the phone user are saved to the local flash file system and are upload with the name *MAC*-directory.xml.

Log files are appended on the server by default until a preset size limit (default 512K) is exceeded at which point the log is truncated on the server. See the *logging* XML tag in the vendor-provided sip.cfg configuration file.

3.3 Special Purpose Analog Line Support

A number of special applications require analog lines. These include

- Elevator emergency phones which require an automatic ringdown when off-hook.
- Security blue light phones. Similar to elevator phones.

- Fax machines and fax servers, some of which use DNIS.
- Other analog instruments such as cordless phones, specialty handsets for visually impaired users and so on.

Analog features required include:

- Distinctive ring for internal and external calls.
- DNIS - Dialed Number Information Service which indicates which number was dialed with DTMF signaling. Used by fax servers and the like.
- Caller ID.
- Emergency failover to a POTS trunk when SIP signaling is not functional.
- FXS for phones.
- FXO for emergency failover trunks.

For these features, we use Quintum AX series rack-mount analog adapters. These ATAs support up to 24 lines and have provision for emergency access to a number of Verizon POTS line should the network or power fail.

Further work on testing and understanding the options for Fax (e.g. T.38 vs. G.711) is needed. Basic faxing over G.711 has been tested however, enhanced capabilities such as Group 3, 33.6 kbps operation, etc. still need to be tested.

4 Call Control

4.1 SIP

Call control is implemented using Session Initiation Protocol (SIP)[1] which is now the mature standard protocol for VoIP, supplanting competing protocols such as SCCP (Skinny), MGCP, and H.323. SIP has a number of components including User Agents, Proxies, Registrars and Location Servers and is an extensible system in which new functionality can be implemented in the end systems without having to change the core components.

Call control is the process of setting up and tearing down sessions (calls) and is implemented in cooperation between UAs (SIP phones) and software that usually combines the functions of Proxy, Registrar and Location Server.

4.2 Registration

SIP Address of Record

SIP User Agents are addressed using a URI scheme similar to that used for email: sip:*user@domain*. This is known as the Address of Record (AOR). The UA registers with the Registrar with this URI and calls to this user are generally addressed to the AOR. Some sample SIP AORs are sip:alice@columbia.edu and sip:+12128541754@siptest.columbia.edu. Two groups of registration name spaces are used by the CUIT implementation and additional name spaces are also accommodated:

Extensions 5-digit phone extension numbers, corresponding to the current internal University PBX dial plan, are used for the AOR in the sip, siptest, and sipdev.columbia.edu domains. A sample AOR is sip:41754@sip.columbia.edu.

University Network IDs UNIs are the unique identifier used at Columbia for email and other network identity purposes within columbia.edu. A sample UNI-based AOR is sip:ac45@columbia.edu.

Subdomains Subdomains of Columbia, notably the Computer Science department, manage their own SIP services. A sample AOR for a user in the Computer Science subdomain is sip:hgs@cs.columbia.edu.

Authentication User

Phones authenticate their registration to the SIP proxy with an authentication user and password using HTTP digest authentication. The authentication user used in the CUIT VoIP system is the same as the SIP AOR user. For example, sip:10501@sip.columbia.edu's authentication userid is 10501. The password is a random value assigned by the provisioning application.

We do not use University Network ID (UNI) for the authentication userid and the UNI password for three reasons:

1. Hard phones are always logged in and available to make calls. As such, the identity needed to authenticate SIP registration really belongs with the extension number and not a specific user. Notably, the same phone number can be used by many users (e.g. for shared phones in public spaces).
2. SIP authentication uses HTTP digest. The UNI password system uses Kerberos. It is not possible to authenticate a Kerberos password using digest authentication since the clear text password is not known to the proxy.
3. Phone security is suspect. We do not know how easy it is to hack the password out of a phone. Furthermore, the password is stored in the central provisioning server in clear text, so a server compromise exposes the passwords.

The web-based Provisioning Tool uses the UNI and Kerberos password through our central authentication system to grant access to modify the SIP user and password.

Locations and Proxies

Phones register with (log in to) the Registrar, indicating their location (IP address and port, known as the Contact) which the Location Server keeps track of. When a call is initiated, one or more Proxies are used to route the call to one or more specific Contacts that have registered with the Address of Record (AOR). For example, the AOR for 12345@sip.columbia.edu might have a specific UA registered with a Contact of 12345@128.59.114.123. The Proxy gets this information from the Location Server and uses it to direct the INVITE to the appropriate UA. Note that several UAs can simultaneously register their different Contacts for the same AOR. This is analogous to bridged phone lines. Calls (INVITEs) to the AOR are generally sent to all Contacts using *parallel forking*.

4.3 OpenSER

The SIP Proxy, Registrar and Location Server software used for the CUIT VoIP service is OpenSER (www.openser.org). OpenSER is a branch of the SIP Express Router open source project (SER; www.iptel.org). SER and OpenSER are highly scalable and used by a large and growing worldwide community. For the sake of brevity, when we refer to the Proxy, we will also mean the Registrar and Location services which are all implemented by the OpenSER software.

Two redundant proxies are implemented, each running on an HP DL360 server running Red Hat Enterprise Linux. The servers are in the Computer Center and Philosophy server rooms. SIP REGISTER messages sent by UAs are replicated between the servers using the openSER t_replicate() function. See figure 1. Selection of which server to contact is made by the UA using DNS NAPTR and SRV record lookups. This provides load balancing of registrations across the proxies and automatic failover. If the currently selected proxy fails to answer, the UA will try the other proxy, which has replicated location data so will work seamlessly after a short timeout delay. Proxies are only involved in call setup, teardown and mid-call signalling (e.g. when a call is placed on hold or transferred). SIP *Record Route* is used so that all call signaling is mediated and tracked by the proxy, permitting implementation of policy and call detail recording (e.g. for charge back purposes).

4.4 Provisioning

Provisioning data (subscriber numbers and passwords, least-cost routing tables, and the like) is stored in a local mysql database on each server. This data is populated by a central provisioning server, described in greater detail in section 11 on page 35. The provisioning system identifies a phone number as the SIP user and then links those SIP users to University Network IDs (UNIs). UNIs can be owners of SIP users or have authority delegated to them by the owner or by CUIT. For example, a Departmental Administrator can be delegated permission to set phone options for numbers in a given department. Multi-line phones can be provisioned with multiple SIP users that belong to different UNIs to implement the common practice of sharing phones by multiple users (e.g. by graduate students) or sharing the same extension across multiple phones in a work group.

5 Media Services

5.1 Asterisk

Media services are voice applications such as voicemail, interactive voice response (IVR), conferencing and music on hold. The Asterisk[3] open source software is well suited to providing these services. Asterisk is a popular soft PBX that supports SIP (as a back-to-back user agent – B2BUA) among other protocols. Its media handling capabilities are excellent and provide extremely cost effective and flexible services such as conference bridging and voicemail running on commodity Linux servers with no special purpose DSP hardware needed.

5.2 Voicemail

The IMAP capability described below is not yet in production. Interoperability with MS Exchange TBD.

Among many of the Asterisk *Voicemail* application features is the ability to deliver voicemail as an email attachment. This application has been enhanced by code development sponsored by UPenn to make the voicemail application into an IMAP client of an email server. This means messages are stored as standard email. The status of messages read or deleted via the email interface is immediately reflected via the telephony user interface and vice-versa. No voicemail is stored on the Asterisk server. However, greetings are stored there. Since these messages are relatively static, simple rsync replication between redundant Asterisk voicemail servers will be used to keep greeting messages in sync. Penn has also commissioned further improvements to the Asterisk Voicemail user interface that are still under development.

The basic Asterisk *Voicemail* system includes “busy” and “unavailable” greetings. The Rolm PhoneMail system has “internal” and “external” greetings. We will substitute these meanings for the busy and unavailable messages. It is expected that further development of *Voicemail* will happen in Phase II, including potentially using Call Processing

Language [4] or VoiceXML[5] to allow customization of answering behavior by time of day, caller ID and other information.

5.3 Conferencing

Three-way conferences are supported by the Polycom phones directly. For larger conferences, the Asterisk *Conference* application will be used. This application does conference mixing using host-based DSP – no special purpose hardware is required. *Conference* supports both a traditional predefined conference which has a given access number and password as well as the ability to create on-the-fly conferences: Simply dial the conference pilot number, pick a conference number and password and then others can dial in to that same number with the given password. A web-based conference management application is available and will be tested as well.

Conference mixing cannot be implemented redundantly since, for any given conference, all callers must connect to the same server. However, for quick recovery and load sharing reasons, it is possible and desirable to have two or more Asterisk conference servers. This number will be scaled up as experience and growth in use of conferencing services dictate. *Conference* servers REGISTER their extension number with the Proxy just like phones do.

5.4 Call Park

Call park support is still being designed.

Asterisk's *Park and Announce* application is used to park a call which puts it on hold in a "parking lot" so that another can dial the pickup extension to retrieve the call.

5.5 Interactive Voice Response

Using general programming capabilities of Asterisk, we are able to implement interactive voice response menus and similar applications as needed. IVR work will be largely deferred until Phase II while we concentrate on rolling out basic service for Phase I.

5.6 Automatic Call Distribution

The Asterisk *Queue* application implements automatic call distribution (ACD). However, due to the fact that we've recently implemented the Siemens Agile ACD application for the computing help desk and Agile, which runs on the small Siemens HiPath 4000 IP-capable PBX, we will implement ACD in Studebaker using Agile and Siemens IP phones. Agile provides a rich management monitoring and reporting interface which is not available with Asterisk. We will re-evaluate the use of Siemens ACD versus an Asterisk or other implementation in the future. A future University decision on implementation of an enterprise Customer Relationship Management system will likely drive this selection.

5.7 Error Recordings

Error recordings (e.g. "You have reached a non-working number") are played using the very simple Asterisk Play application. Special extensions are defined which OpenSER forwards calls to for the various error cases.

```
mailbox (or # for own number)
password
You have x new (and x old) message(s)
You have x old messages
press
1 - for new (old) msgs (only if there are some)
    plays each message
    3 - advanced options
        1 - send reply (doesn't work)
        3 - hear message envelope
        * - return main menu
    5 - repeat current message
    7 - (un)delete
    8 - forward msg
        extension?
        1 - prepend message
        2 - forward w/o prepending
    9 - save this message
        which folder?
0 - new msgs
1 - old msgs
2 - work msgs
3 - family
4 - friends
        # - cancel
    * - help
    # - exit
2 - change folders (and then back to main)
    0 - new msgs
    1 - old msgs
    2 - work msgs
    3 - family
    4 - friends
    # - cancel
3 - advanced options
    (no options???)
    * - return to main menu
0 - mailbox options
    1 - record unavail
    2 - record busy
    3 - record name
    4 - record temp grtg
    5 - change password
    * - return to main menu
    # - repeat menu
# - exit
```

Figure 8: Asterisk Voicemail Menus

Extension	Purpose
99001	Blocked Caller ID not accepted.
99002	Do not disturb (and no voicemail available)
99003	Busy (and no voicemail available)
99004	Unavailable (and no voicemail available)
99005	Non-existent number

6 Development, Test, and Production proxies and media servers

6.1 Server Platform

All VoIP servers are identical commodity Intel Linux servers (currently HP DL360 running Red Hat Enterprise Linux Release 4). Development is performed on one dedicated HP DL360. New code releases, etc. are developed on this server before being moved to a pair of test servers. Once testing is completed, the new releases are rolled out to six production servers: two OpenSER proxies, two Asterisk voicemail and two Asterisk conferencing servers.

The test servers are a close approximation of the production Proxy server configuration, notably being a redundant pair, although not needing to be located in separate data centers. Asterisk media service can run on the same host as the test servers but may be run on the development server if necessary to more fully mimic the production environment.

6.2 Routing of PBX-originated calls to dev/test/prod

For testing that requires inbound calls from the PBX, specific DID numbers are redirected to the development or test servers. All other numbers go to the production servers. DID testing has not been completed with ITSPs so procedures have not yet been established for directing particular number ranges to development, test or production proxies.

extension range	server destination
re-homed user extensions 10210-19	production
10500-49	development
10550-99	test

6.3 Server Configuration Management and System Administration

The servers and Linux OS are provisioned and managed by the CUIT Technology Infrastructure Unix and Email Systems Group, using the same methods used for about 200 other CUIT Solaris servers and about 40 other CUIT Linux servers including those supporting the 80,000 user Cyrus email system. Per the standard CUIT Linux management methodology, server functions are defined by the *hostmonger* host management system which is a component of the automated configuration management environment. Hosts are defined to be members of hostmonger *clusters* as shown in table 1.

A Linux host when initially installed (or re-installed) uses PXE boot to load an initial kernel image and runs kickstart which installs cfengine. [6] The cfengine configuration uses groups defined in hostmonger to select the required packages for OpenSER, Asterisk, and so on. In about an hour after being racked, a brand new host can be up and running with a complete configuration.

sip	proxy voicemail conference presence	dev test prod

Table 1: Naming scheme for VoIP project host clusters

name	**location**	**environment**	**use**
jello	CC	test	proxy, voicemail, conference
tiramisu	CC		proxy, voicemail, conference
mousse	CC	dev	proxy, voicemail, conference
torte	CC		presence
eclair	Phi	prod	proxy
cake			voicemail, conference
sherbet			presence
cocoa	CC		proxy
pie			voicemail, conference
sorbet			presence

Table 2: VoIP Servers and Services

Specific servers and their uses are listed in table 2.

6.4 OpenSER and Asterisk Configuration Management

Configuration files are managed with the same methods used for other CUIT Unix systems: They are maintained in the source code tree (in /src/systemfiles/etc/openser and /src/systemfiles/etc/asterisk, respectively) using the Revision Control System[7]. RCS keeps a history of all changes as they are checked in, permitting easy determination of what changes were made and by whom, reversion of a change as needed, and branching and later merging between development and production branches.

Since configurations are largely identical for each server but need some per-host customization, the source files are managed with the C pre-processor, CPP, using #ifdef–#else blocks. The CPP symbols that are tested are defined by the hostmonger tools which include a wrapper around CPP that defines the appropriate cluster symbols on a per-host basis.

The generated per-host files are then compared (with diff) to copies of the most recently installed version. This process is driven by a Makefile, which has a “make” target and a “make install” target which pushes the configurations out to the target hosts.[1]

[1] CUIT is in the midst of implementing cfengine so some file update functions are managed by cfengine while others are managed by the hostmonger tools. Make install, could, for example, either directly install the files on the managed hosts (via a hostmonger wrapper around scp) or install them in the cfengine repository from which they would be pulled by the hosts.

7 Telephony Services

There are a number of services that are implemented by a traditional PBX that customers expect to work with SIP hard phones. Unlike a PBX, where features are generally implemented centrally and phones are relatively dumb devices, most SIP features are implemented directly by the phone. Others take extra support from, for example, Proxies or Presence servers. Many features can be implemented either centrally or by the phone. In as many cases as possible, to enhance scalability, when a service could be implemented centrally or by the phone, we chose to use the phone's implementation.

7.1 Business PBX Features

The basic SIP protocols implement 90% of the features needed for a VoIP system that replaces a business PBX. However lack of a number of those features – the other 10% – can be show-stoppers when it comes to user acceptance of the new system. Many of these business features have been implemented in proprietary ways by some VoIP vendors. Polycom has adopted the approach of Sylantro.com, which uses open, published SIP protocols (final standards as well as draft proposals). As such, it is possible (although sometimes difficult) to implement business PBX features[8, 9] such as:

Directed and group call pickup

In directed call pickup, an individual can answer an incoming call (e.g. heard ringing down the hall) using his or her own phone. This is typically implemented by entering a pickup code followed by the extension of the ringing phone. In SIP, this is implemented using the Replaces header[10] to "steal" the incomplete dialog INVITE.

Group call pickup is analogous to directed pickup with the difference being that one enters the group pickup code (or key) to answer any ringing phone in the work group that one is a member of.

Call park and pickup

In call park, a callee REFERs[11] the call to a parking lot (which may optionally play music on hold). Call pickup re-INVITEs the call from the parking lot to the intended recipient using the Replaces header.

Bridged line appearances (BLA)

Bridged line appearance (BLA; also known as multiple or shared line appearance) [12, 13, 14] implements a boss-secretary/receptionist relationship typically seen with key systems. Two or more phones share the same extension and have indications of the status of the shared line such as solid red for off-hook or in use and flashing red for on hold. Furthermore, the on-hold line can be picked up from any phone that shares the extension and indicates flashing red. This capability is implemented with a Presence server and user agent that SUBSCRIBEs to the "dialog" and "sla" state of each phone. Each state change on the phone results in a NOTIFY which is distributed to other subscribers (phones).

Distinctive ringing

Distinctive ringing is implemented with the SIP Alert-Info header. The OpenSER Proxy marks the header with a different ring type to distinguish between internal and external callers.

Intercom

Buzz Intercom also uses the Alert-Info header to cause the phone to ring once and then answer. Intercom groups are implemented in the Proxy to restrict who may intercom whom.

Calling and called party display

Calling and Called party display were originally implemented with the Remote-Party-ID header whose draft expired. This capability still works on most UAs. The final standard replacement uses the P-Asserted-Identity[15] header.

Centralized conferencing (for more than 3 parties)

Centralized conferencing support is still being designed.

The Polycom phone can be configured to use a centralized conference service in lieu of the built-in 3-way capability. This enables larger conferences, analogous to the Rolm ability to include 8 parties in a basic conference.

Call forwarding indication

Call forwarding indication is still being researched.

When one receives a call and then forwards it to another – either unconditionally ("call forward always"), or after answering ("transfer") – it is useful to the final recipient of the call to have an indication on the phone display of the caller ID of the original caller and that the call was forwarded, including the caller ID of the forwarder. The Polycom phone might use the Referred-By header[16] to indicate this.

Polycom support of Sylantro features

The Polycom phone supports most of the Sylantro features in their phones. Sylantro and others have documented many of the required business PBX feature implementations. This means we will be able to build or acquire the components necessary to implement them. We've already implemented several of the above features (distinctive ring, intercom, calling and called party display, and bridged line appearances) and are working on the others (centralized conferencing, call park and pickup).

IETF BLISS working group

The concern for vendor interoperability for business SIP features led to the creation of the IETF Basic Level of Interoperability for SIP Services (BLISS) working group at the April 2007 IETF meeting based on a draft requirements

document[17] first published in February 2007 and the latest (March 2007) draft of the BLA specification[12]. We are tracking developments in this working group.

7.2 Phase I Features and Services

Since it is difficult (or sometimes impossible) to implement certain PBX features, we've chosen to not implement those and document a substitute or defer implementation.

Phone services are summarized below.

Phase I Features and Services		
Feature	**Status**	**Notes**
Multiple Line Appearances	complete	Up to 2 on the Polycom 430. Up to 6 on the Polycom 601 to a maximum of 48 using sidecars.
Bridged Lines	partial	RFC3261 SIP parallel forking for inbound calls. RFC3265 event notification using the RFC4235 dialog event package [13, 14, 12] for busy lamp, hold, and pickup features. Pending provisioning tool support.
Privacy	complete	Standard feature of SIP forking. Having the ability to do Barge (Executive Override) is more difficult to implement. See section 7.1.
Multiple Lines per Registration	partial	The same registration (extension) can be assigned to multiple stacked line keys. When the first line is in use, the second starts indicating on a new incoming call and so on. Like a hunt group but without using multiple extension numbers. An alternative to Call Waiting. Implemented in phone. Pending provisioning tool support.
Call Waiting	complete	Up to 8 waiting calls per line are supported.
Caller & Called number & name display	complete	Calling name/number use the From and/or Remote-Party-ID[18] headers in the SIP INVITE. Called name comes from the Proxy inserting a RPID header in the 180 or 183 response to the INVITE. Use of RFC3325 P-Asserted-Identity will be implemented to replace the RPID which never made it out of draft.
Speaker	complete	Full duplex, available on all models. Polycom's claim to fame is the quality of their full duplex speaker phones.
Headset	complete	RJ11 jack on the back of the phone and a headset select button. External interface box and handset lifter not needed.
Cordless Headset	pending	Testing of Bluetooth cordless headsets is needed.
Mute	complete	
Hold	complete	RFC3264 [19] method.
Transfer (Consultative, Blind)	partial	RFC3515 [11] REFER method. An unexpected caller ID is displayed on the transferred-to phone: The identity of the phone that initiated the transfer rather than that of the original caller is shown.

continued

Phase I Features and Services		
Feature	**Status**	**Notes**
Call Forward (Always, Busy, No Answer)	partial	Set by Provisioning Tool and phone, which has a visual forwarding indication. Standard forwarding is to voicemail "busy"/"unavailable" entry point. User training will be required to determine when one would use system call forwarding – which forwards an extension which may have multiple phones sharing it – and when to use station call forwarding which only forwards a particular phone. In all but the case of bridged lines, these are equivalent.
Do Not Disturb (DND)	complete	Proxy implements unconditional forwarding but phone also has a DND function and visual DND indication.
Redial	complete	Standard Polycom feature.
Call Return	complete	See Call Log, below.
Call Log	partial	Phone has a menu of missed, received, and placed calls. Includes ability to dial from this menu. Web-based listing to be implemented as well.
Buzz Intercom	partial	Proxy sets RING_ANSWER RFC3261 Alert-Info header in INVITE which phone interprets to ring once then answer. We use a Triplet ring tone. Implemented in Proxy. Pending Provisioning Tool support.
Intercom groups	partial	Permissions for who may make an Intercom call. Implemented in Proxy. Pending Provisioning Tool support.
Hunt Groups	testing	Using OpenSER LCR serial forking implementation.
Call Park/Pickup	pending	See Asterisk *Park and Announce* and [20] use of RFC3515 Refer and RFC3087[21] "callpark". See section 7.1.
Directed Call Pickup	pending	See section 7.1.
Group Call Pickup	pending	See section 7.1.
Message Waiting Indicator (MWI)	complete	Sent by Voicemail using SIP NOTIFY of message-summary per RFC3842 [22] (using implicit SUBSCRIBE based on registration). On shared multiple line appearances, the overall phone indicates MWI with a blinking message light and stutter dialtone plus each line appearance individually shows a message waiting (envelope) icon.
3-Way Conference Calling	complete	Phone can mix itself and two other parties.
Multiparty Conference Calling	pending	See section 7.1.
Station Speed Dial	complete	Uses phone directory.
System Speed Dial	pending	Uses OpenSER speeddial function. Pending Provisioning Tool
Distinctive Ring	partial	Proxy sends Alert-Info header in INVITE for external vs. internal calls. External calls will be indicated by a Double Trill ring and internal by a Single Trill. Phone also supports personal distinctive ringing based on the directory but phone settings are overridden by the Alter-Info header. Pending provisioning tool support.

continued

Phase I Features and Services		
Feature	**Status**	**Notes**
Personal Security Codes (PSC)	partial	Allow charging calls on a restricted line by prefixing the call by dialing 97+10-digit security code. Pending operationalizing transfer of PSC database from Rolm to VoIP.
DTMF passthru	complete	RFC2833 [23] method used to pass DTMF digits through.
Anonymous Call Reject	complete	Set via Provisioning Tool.
Anonymous Calling	complete	Using INVITE From and Remote-Party-ID headers. Not truly anonymous except after crossing the SIP-to-PSTN gateway since Contact headers are required by SIP. (We are not implementing a call anonymizer service.) All-call restrict/unrestrict is set in the Provisioning tool.
Restricted Calling	complete	Explicit permissions for on-campus, inter-campus, local, domestic long distance, international, and pay-per call are set in the provisioning tool.
Conference Bridge (Scheduled, Ad hoc)	development	Asterisk *Conference* in conjunction with phone central conference server features.
Analog Lines	testing	See section 3.3, on page 14, above.
ADA Compliance (hearing & visual)	complete	Will be handled on a case-by-case basis, using analog instruments where necessary
Voicemail	partial	Uses Asterisk Voicemail application. IMAP client support is being tested.
Voicemail Forwarding	complete	Forwarding among Voicemail users is possible. Forwarding to Rolm Phonemail is not. See section 15.
Voicemail Broadcast	partial	Broadcast voicemail announcements. Will be implemented using the Cyrus email system to send an email broadcast with the voice attachment. Currently implemented as a number that is dialed to record the message and a script that pushes the message to all voicemail users.
Voicemail Distribution groups	pending	User-specified distribution lists for voice mail.
Voicemail Referral Extension	complete	Dialing '0' transfers to a referral extension.
Automatic Call Distribution (ACD)	testing	Implementing using Siemens Agile ACD and Siemens IP phones.
E911	development	See section 8 on page 27.

Deprecated Features and Services	
Feature	**Notes**
Executive Override/Barge	Not used in practice. New Call waiting display meets the need.

New or Improved Features in Phase I	
Feature	**Notes**
Call Waiting	See phase I description.
Multiple Lines per Registration	See phase I description.
Call Log	See phase I description.
Redial	Redial is supported on the Rolm but one must remember to hit SAVE/RPT prior to hanging up. Polycom phone redial is "normal."
Voicemail as Email	Voicemail email integration is a major improvement.
Web access to provisioning settings.	

7.3 Phase II Features and Services

In Phase II of the project, we intend to roll out many features beyond those of a basic PBX replacement. In Phase II, the power of SIP to support more advanced real time communications will become apparent.

Planned Phase II Features and Services	
Feature	**Notes**
Group Ring	Ring all phones in a group.
Camp On	Camp on a busy line and call it when available.
Downloadable ring tones	Use of Alert-Info header and/or phone download.
ACD Agent	This is subject to future support by Siemens Agile ACD (not currently supported for non-Siemens phones) or other future ACD product support of Polycom SIP phones.
Automated Assistant	Dialing by name, etc.
Voicemail Name Directory	Voicemail supports forwarding messages by the first three letters of the recipients last name. TBD.
Voicemail Groups	Personal groups of voicemail recipients.
Fax email routing	This could replace the need for departmental fax servers.
Signaling and Media Encryption	Use of SIP over TLS (sips) and Secure RTP [24].
Interactive voice response.	Asterisk is well-suited for this.
Programmable user services	Use of CPL [4], investigation of VoiceXML, etc.

8 911 and Emergency Calling

8.1 Legacy PBX 911 Implementation

Columbia has an emergency number (dial x99, 4-5555, or 212-854-5555) which rings the Public Safety command center. In addition, 911 (dial 93911, 9911, 911) is routed directly to the NYC 911 PSAP, bypassing Public Safety. Public Safety receives no indication that a 911 call has been made.

911 calls are routed to conventional Verizon POTS lines (*not* CAMA trunks) and indicate the address of the call as 2960 Broadway, which is the location of the PBX in Low Library. Caller ID (calling number) is not provided. All 911

calls served by the main 9751 model 70 PBX for the area spanning the Morningside campus and nearby buildings, covering approximately 113th to 122nd Streets between Morningside Drive and Riverside Drive, indicate the single address.

99 (4-5555) calls are routed to Public Safety and provide a calling name and number display. No indication of location is provided. (Public Safety is currently in the process of evaluating their command center operations with a consultant and part of the scope of this project is a Computer Aided Dispatch system that will improve this situation.)

For the eleven remote 9751 model 10 and 50 PBX systems, Verizon POTS lines are similarly used. Since these other PBX systems are usually in a single building, 911 calls indicate the address of the given building, but not floor or room number.

8.2 Initial VoIP 911 Implementation

The planned initial VoIP 911 implementation will contract with an ITSP for 911 service. Traditional number-based static addresses will be provided in Master Street Address Guide (MSAG) [25] format. We will FTP updates nightly. This will be sufficient for the initial VoIP implementation in Studebaker as the phones will all be tied to desks and not likely to be relocated. We audit Ethernet switch MAC address data nightly to detect a phone that has been relocated.

8.3 Intermediate VoIP 911 Implementation

In our intermediate 911 implementation, we will likely contract with Intrado (or one of our contracted ISTPs) to provide 911 service. UNC is currently working with Intrado and we plan to leverage this work. Intrado is one of the major national 911 service providers used by VoIP providers like Vonage. Intrado is able to properly route 911 calls to every PSAP nationally. Location database updates are somewhat more dynamic and we expect to enhance our current network location database tools to track phones based on their IP address, router ARP caches and switch MAC address tables. See figures 9-11 for an example of this process.

8.4 Future VoIP NG911 Implementation

We expect future SIP User Agent implementations to use DHCP [26] or LLDP-MED [27] to learn their wired location and to send it as part of the INVITE for a 911 call. Wireless devices will need to triangulate their location (or the wireless infrastructure might perform this function).

The NG911 Project [28, 29] at Texas A&M and Columbia is defining the next generation 911 system that will handle worldwide mobile emergency calling. These developments are still several years away from implementation, with initial standards expected to be finalized in early 2007. We will track their development and implement as they become available.

9 PSTN and Legacy Rolm PBX Connectivity

Connectivity to the PSTN is accomplished through SIP trunking service contracted from two or more carriers. At present, we have outbound domestic and international service from Qwest and in- and out-bound local, domestic and international service from PAETEC (under test). Legacy TDM carrier connections on multiple PRIs are in place to the Siemens Hicom 300 PBX.

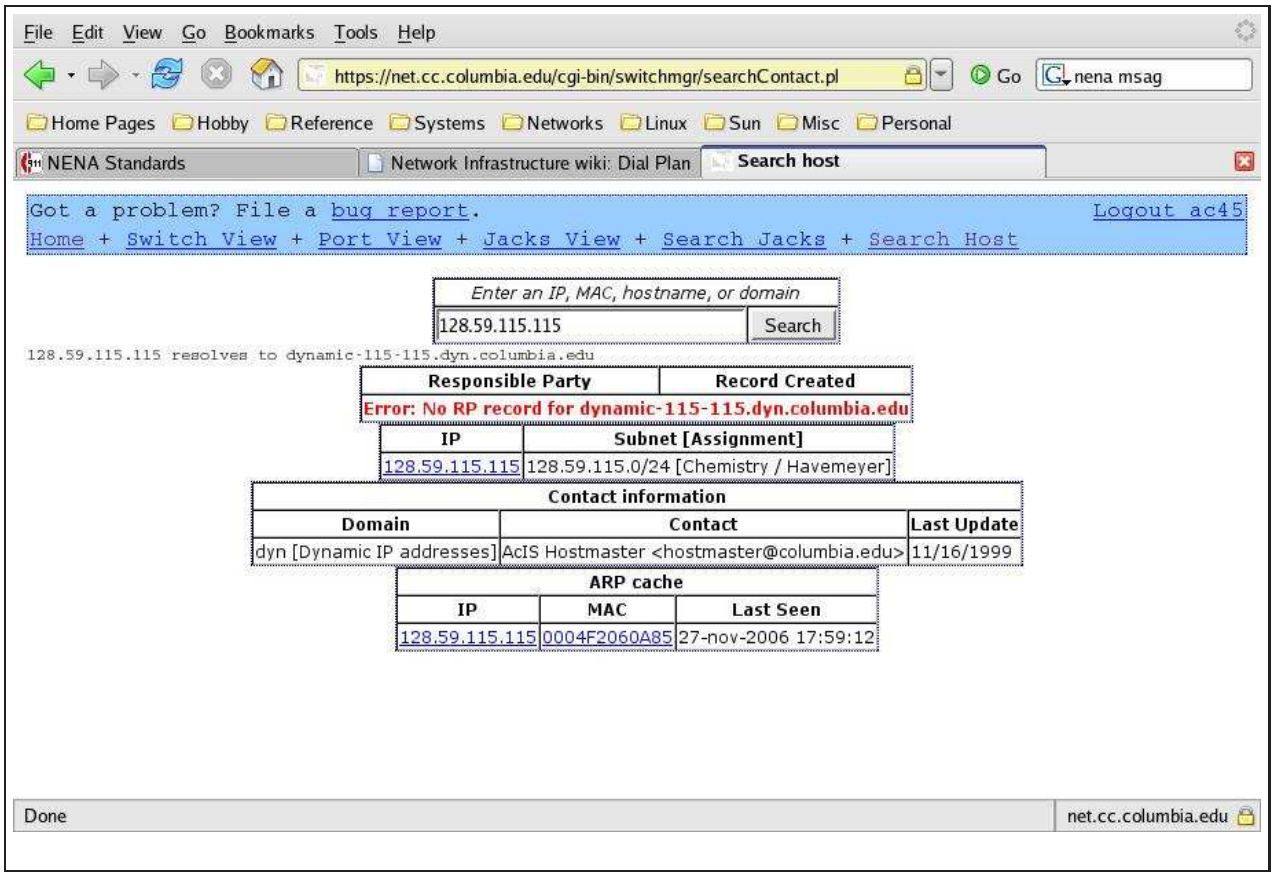

Figure 9: Switchmgr tracking of an IP address: IP to MAC mapping

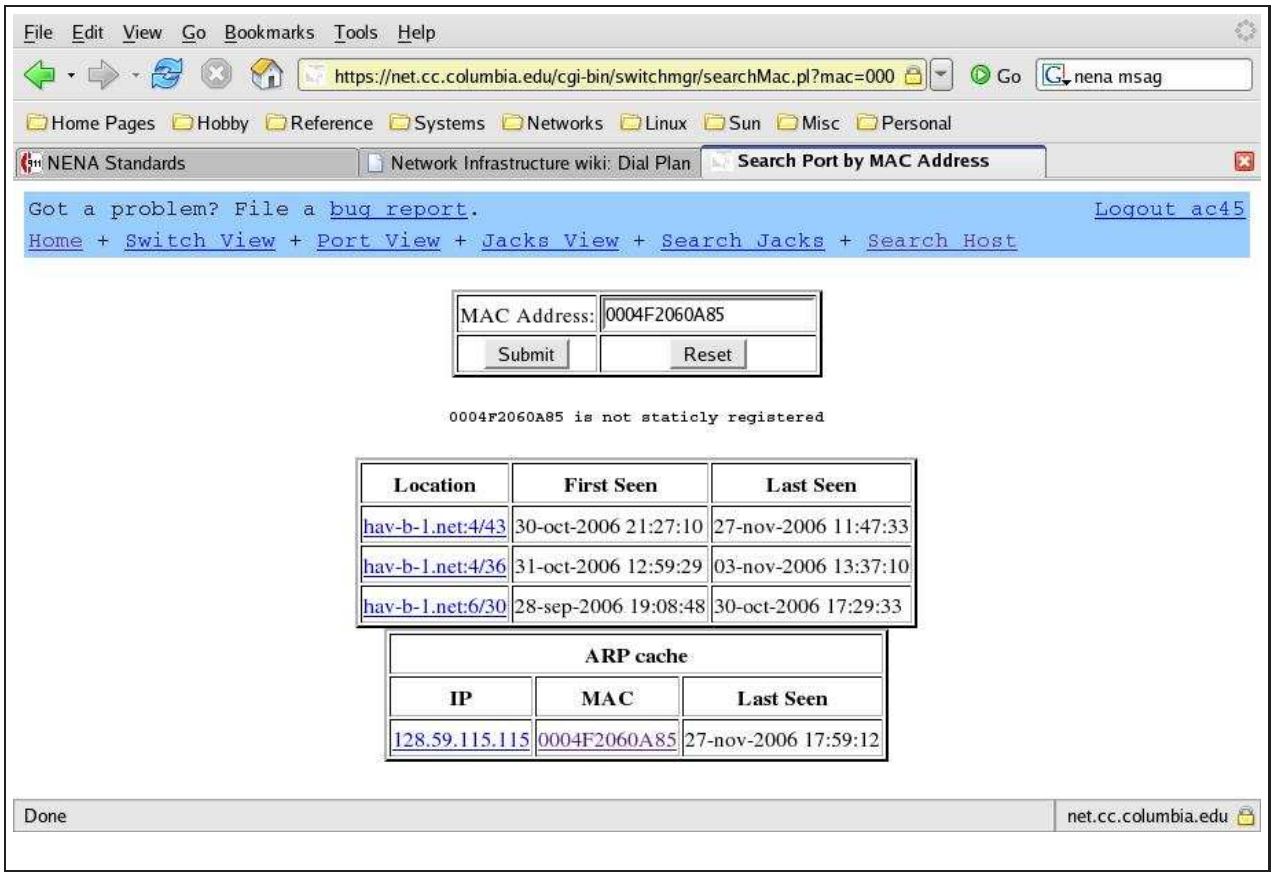

Figure 10: Switchmgr tracking of an IP address: MAC to switch port

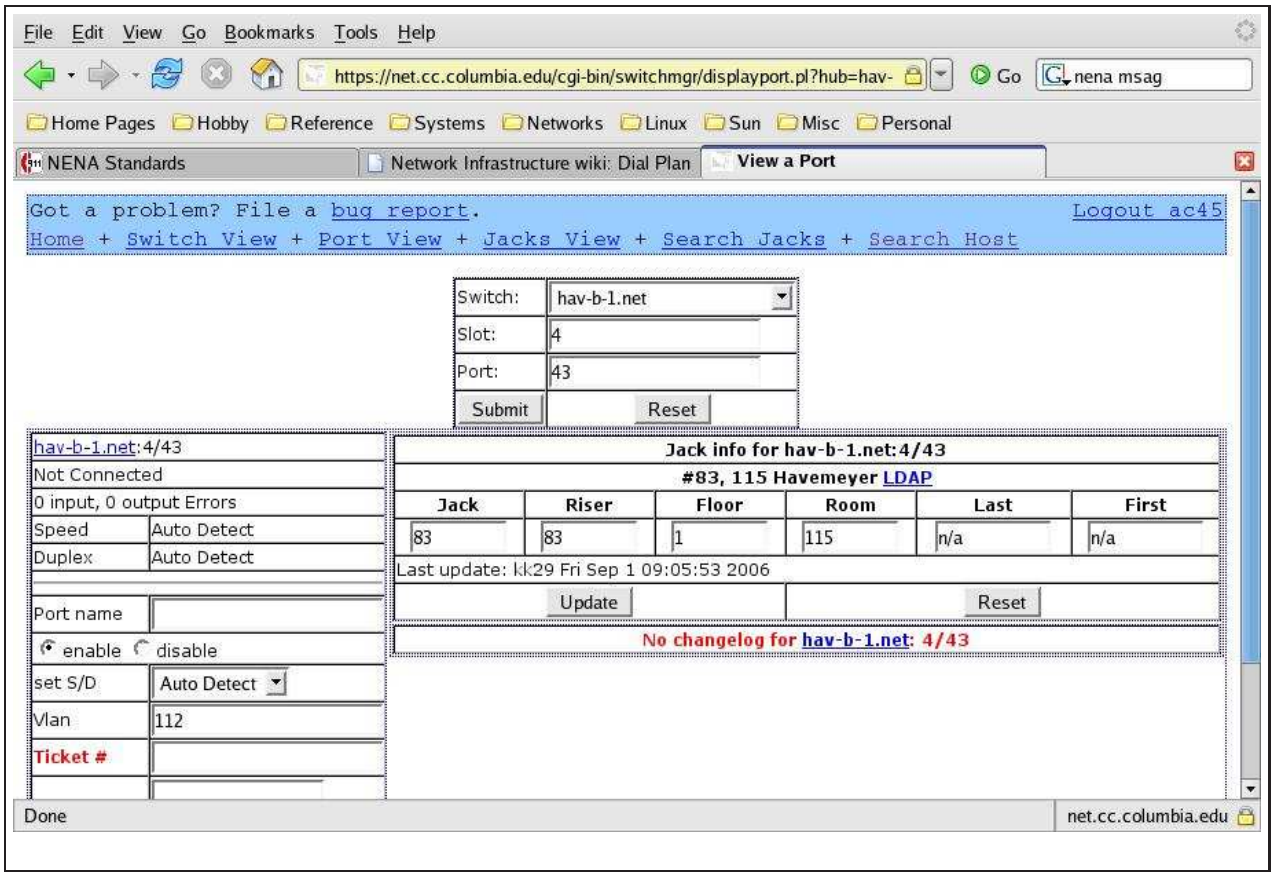

Figure 11: Switchmgr tracking of an IP address: Switch port to jack

9.1 ITSP SIP Trunking

SIP trunking implementations vary by carrier. Some require a complete E.164 number in Request URI (e.g. +12128541754). Others want only 10 digits (2128541754), others 11 digits and so on. Due to these differences, all Request URIs are rewritten in the Proxy into standard E.164 format. For example, a domestic long distance call is written as +12135551212 and an international call as +44123456789. The OpenSER Least Cost Routing (LCR) tables are used to select carriers by dial prefix, specified priority and then random order. Entries in the LCR table specify the number of digits to strip and the prefix to insert. For example, one ITSP requires international calls to start with 011, so the example above would strip one digit (the +) from the E.164 number and then prefix it with 011.

In order to provide proper Caller ID, some ITSPs accept the Remote-Party-ID[18] header, rewritten in E.164 form. Other ITSPs also require rewriting the From and even the To headers into E.164. Since From and To headers cannot be modified in a dialog [1, Sect. 8.2.6.2] this requires the Proxy to have to rewrite outgoing INVITE From and To headers and to restore the headers on the incoming responses before they are delivered to the phone. (It is not clear why the To header would ever need to be rewritten, as the Request URI contains the destination number.) The openSER *uac* module is used to modify and restore the From header in a transaction.

When a call is rejected (403, 5xx or 6xx) or times out, the failure_route handler picks the next available gateway from the LCR table.

To ensure good QoS, a direct carrier IP peering is desirable, using alternate IP carriers as a backup. In practice, routing over the commodity Internet has worked quite well. We continue to investigate peering options as well as SIP service from our primary ISP, Broadwing (recently acquired by Level 3).

9.2 TDM Trunking

TDM trunking, which is used primarily to connect to the Rolm PBX but could potentially also be used to swing current TDM carrier lines from the PBX, is accomplished using two Cisco 3845 Integrated Services Routers (media gateways) connected to the HiCom 300 via ISDN PRI lines (see figure 1). To OpenSER, these media gateways look just like those of ITSPs. In fact, using LCR priorities, ITSPs can be used to route intra-campus (SIP to Rolm) calls in the event that both 3845 media gateways or their associated PBX interfaces fail. Conversely, outbound SIP calls can be routed via the Rolm PBX if ITSP proxies fail.

The PRI interfaces use ISDN Q.931[30] signaling which includes caller ID (CLID) and calling name (CNAM) Information Elements. See section 15, below, for more details on interoperability issues.

9.3 Routing and porting of 5-digit extensions

A key feature of the CUIT VoIP implementation is that customers retain their 5-digit campus extensions when moving from one of the several PBXes to the VoIP system (and back again). These systems include:

- Morningside campus Rolm 9751 model 70
- 1700 Broadway Rolm 9751 model 50
- Miscellaneous other Rolm 9751 models 10, 50
- 330 Fifth Avenue Intertel system.
- Verizon Digital Centrex service.

In all cases, the interface for routing between the VoIP system and the other PBX will be via the Morningside Rolm 9751 and associated Siemens HiCom 300.

9.3.1 VoIP to Rolm routing

VoIP to Rolm routing is accomplished by checking if the extension is in the rolm_extensions table. (This is an extract of data from the Rolm provisioning database.) If so, route using LCR to the Cisco 3845's or use a backup route via an ITSP for DID numbers (851,853,854 but not 7-xxxx).

9.3.2 Rolm to VoIP routing

The Rolm 9751 REXT (remote extension) command and the Siemens HiCom 300 DPLN (dial plan) command are used to individually route each VoIP extension (or block of numbers) to one of the trunk groups associated with the Cisco 3845 media gateways.

9.3.3 Porting between Rolm and VoIP and vice-versa

Porting an extension from the Rolm to the VoIP system involves:

- Add it to the OpenSER *subscriber* table in each proxy.
- Point the extension from the Rolm 9751 to the HiCom 300 with the REXT command.
- Point the extension from the HiCom 300 to the media gateway trunk group with the DPLN command.

A "still on Rolm" flag is implemented in the Proxy to allow newly-provisioned VoIP phones to be installed prior to cutting over from the Rolm, which is accomplished by simply changing the flag in the Proxy.

Porting back reverses the above procedure.

9.3.4 Porting between 330 Fifth Ave Intertel and VoIP and vice-versa

This process adds a step. In addition to porting the numbers via the HiCom 300, they must also be re-pointed in the Intertel PBX at 330 Fifth so that local callers there are routed to the VoIP system.

XXX - need details

9.3.5 Porting between Verizon Digital Centrex and VoIP and vice-versa

XXX - need details

9.4 Dial Plan

The VoIP dial plan is based largely on the legacy Rolm PBX dial plan. Routing and other features are implemented in the OpenSER proxies. The Polycom phones are also configured with a dial plan to provide features such as "outside" dialtone, immediate dialing upon entering the appropriate number of digits and dial completion timeouts for international calls where the number of digits can vary.

XXX - update this table

Prefix Pattern	Use	Notes
1xxxx 3xxxx 4xxxx 7xxxx	5-digit extension	Can be on VoIP or legacy PBXes. Translate to E.164: +1-212-85x-xxxx except 7xxxx which are internal extensions.
2x 2xx	Intercom	Dial 2 or 3-digit Intercom group number.
51xxxxx	CUMC tie line	
90	CU Operator	
94xxxx	TC tie line	
95xxxx	LDEO tie line	
981	Barnard residence halls tie line	
911 93,911 9911	911	Calls NYC PSAP
93,311 93,411	x11	NYC info, directory info, etc.
93,1xxxxxxxxxx 1xxxxxxxxxx	North American number	Comma indicates secondary dialtone on Polycom phone. For convenience also allow 11-digit NANP number without 93 prefix.
93,011x...	International number	variable length; wait 3 seconds and dial
97,ppppppppppp,1xxxxxxxxxx 97,ppppppppppp,011x...	Personal Security Code	PSCs are used to make calls from restricted lines.
99	CU Emergency	Calls Public Safety command center x4-5555

10 SIP Internet Connectivity

Phase I of the VoIP service focuses on PBX replacement and PSTN calling and is largely an intranet SIP service. Internet SIP-to-SIP calls are permitted by the system design but are not expected to be a significant portion of call volume. In later phases of the service roll out, CUIT will focus more on Internet SIP-to-SIP calling and advanced applications enablement. Some areas planned include:

- Interconnection with CU Computer Science SIP system.
- Presence and instant messaging.
- ENUM[31, 32]–based routing of calls.
- Use of email-style URIs (e.g. sip:alan@columbia.edu) and mapping them to DID numbers.

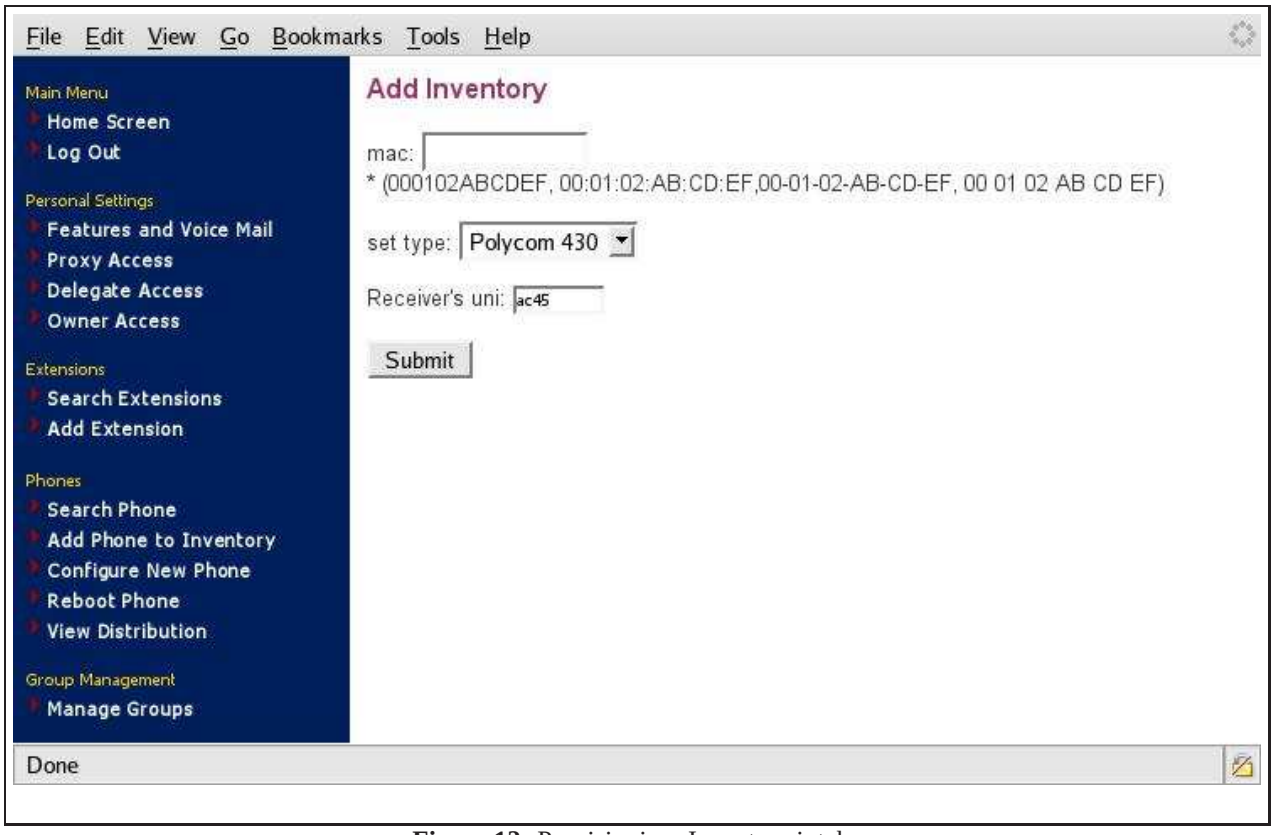

Figure 12: Provisioning: Inventory intake

11 Provisioning and Workflow Management

Provisioning consists of:

- Configuration of SIP subscriber information that is used by the OpenSER Proxies. This includes userids (extensions), passwords, voicemail features, and other privileges (e.g. intra-campus calling, local calling, long-distance, international, pay-per-call 900 numbers, etc.).
- Configuration of voicemail subscriber information that is used by Asterisk.
- Configuration of individual IP phones, including binding them to a customer and adding SIP registrations (line appearances), AES key, HTTPS user and password.

11.1 Web Provisioning Tool and Deployment Workflow

The provisioning system presents a web interface to CUIT Design staff and consists of an underlying series of Perl scripts that modify data in the master provisioning database and pushes it out to the OpenSER Proxies, Asterisk media servers, and Polycom phones. The system is based on code written by Penn which they've shared with the consortium.

The provisioning system is populated with data from the Rolm and VoIP systems and supports the following workflow:

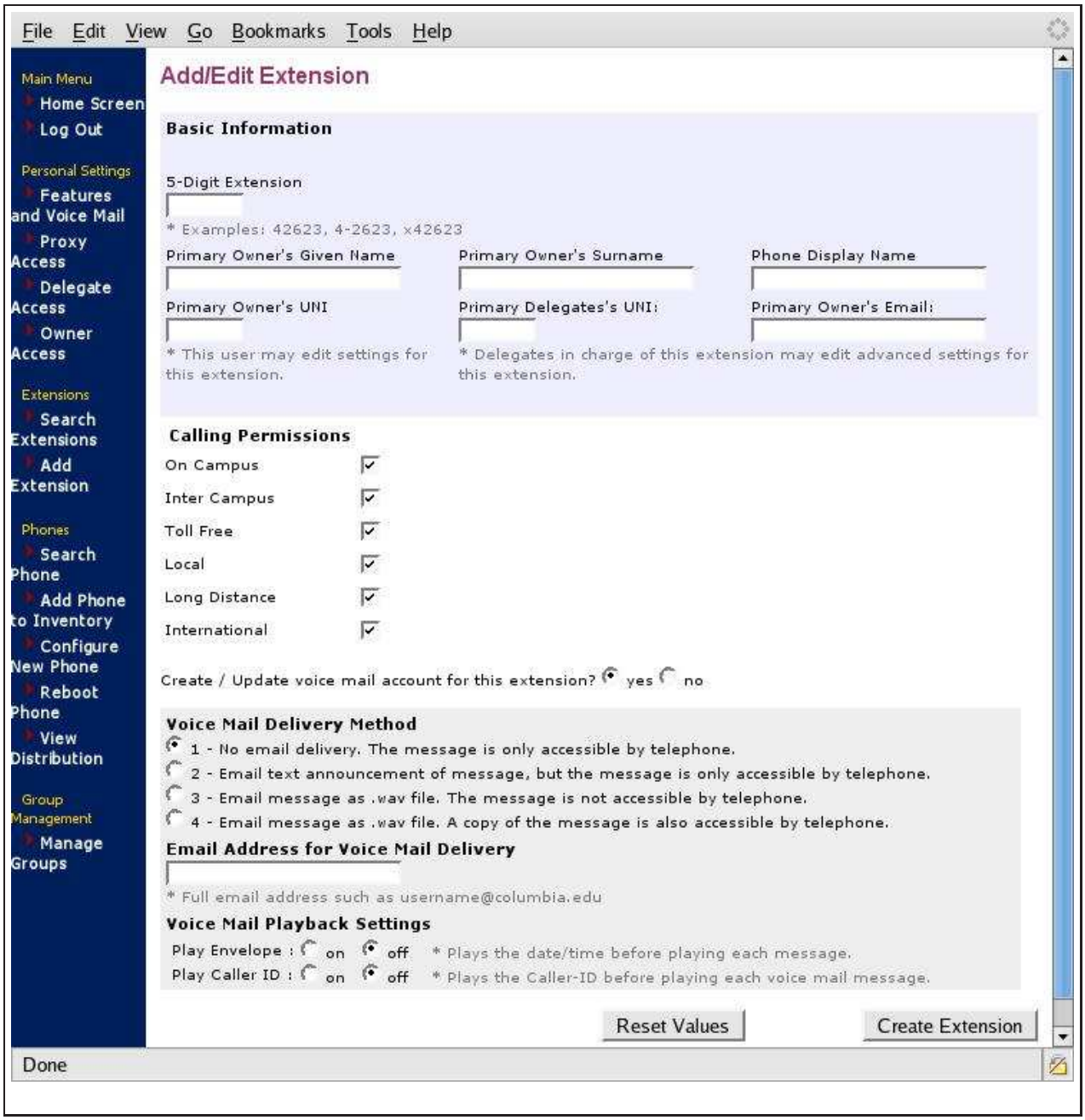

Figure 13: Provisioning: Create SIP registration

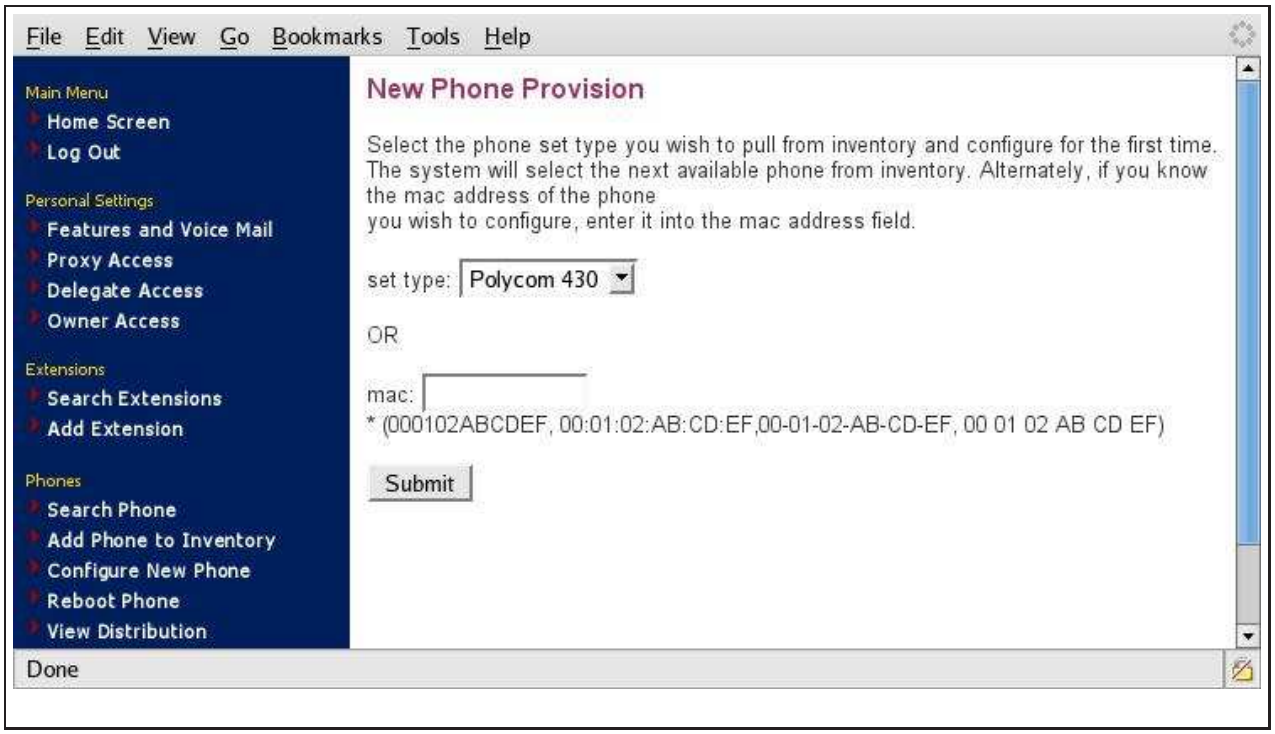

Figure 14: Provisioning: Allocate a phone from inventory

- Intake of a Polycom phone upon receipt in the stock room. The phone's MAC address is scanned with a barcode reader and it is plugged into the provisioning subnet (figure 12). This causes the phone to go through several cycles of reboots to upgrade the boot loader and SIP application, and to install the encryption key and bootstrap "unprovisioned" SIP registration with extension 77999. Our vendors have indicated that they will provide the MAC addresses electronically when shipped so we also have the option of bulk loading MAC addresses in this manner.
- Creation of a VoIP SIP registration (figure 13).
- Select the next Polycom phone from inventory or use a specific one by entering the MAC address (figure 14).
- Assign previously-defined extension(s) and note the phone's intended location (figure 15). Once this step is completed, the phone can be rebooted from the application which finds the phone's registration to OpenSER and sends a NOTIFY check-sync.
- If the phone extension is being ported from the Rolm PBX, then a switchroom task is scheduled to make the necessary 9751 REXT and HiCom 300 DPLN configuration changes. For remote PBX sites, these changes can disrupt service to other customers as extension ranges generally have to be broken apart so this work is scheduled for off hours. For main campus PBX customers, there is no disruption as the phone is a direct extension.
- The phone is handed off the Network Field Services and/or Desktop Support to deliver to and install at the end user location.

The Provisioning Tool also allows setting user options, described below, and can search for an existing phone by MAC address, registered extension or customer name. Once found, the OpenSER proxies are queried to find the phone's current Contact address so that it may be rebooted remotely after configuration changes are made.

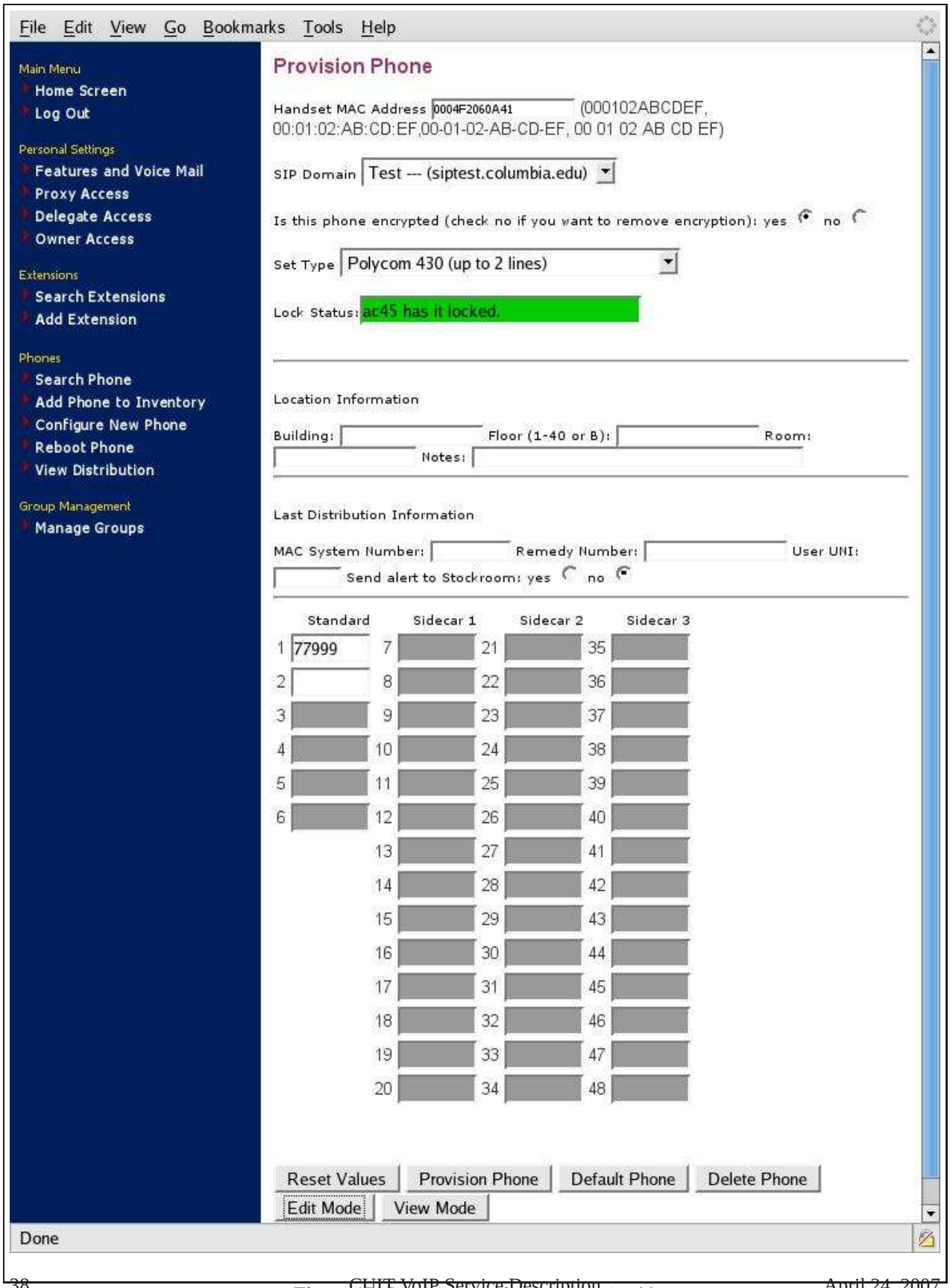

Figure 15: Provisioning: Assign extension(s)

12 Provisioning Tool User Options

The Provisioning Tool is based on Penn's "My iPhone". It allows the owner of the phone or a number of delegates to make configuration changes to features like:

- Call forwarding destination.
- Do not disturb.
- Voicemail password changes and message receipt options.
- Delegate selection.

Many features like forwarding and DND are also available as phone features. The interaction of centrally managed and phone-based features needs to be carefully understood and balanced.

13 Billing and Carrier Bill Reconciliation

CUIT uses a legacy mainframe billing system, WCS, coupled with a number of homegrown tools that pull CDR from the dozen or so Rolm switches we maintain (central 9751 Model 70 plus several satellite Model 10s and 50s). OpenSER CDR is written to the local mysql database on each proxy. Periodic cron jobs will pull the CDR data together into the master billing database from where it will be fed into the WCS system.

Carrier provided CDR is correlated with Proxy CDR using the same tools used for legacy carrier CDR. We have identified the need for the SIP unique Call-ID added to the carrier CDR to aid this process and have communicated this need to the carriers.

CUIT is implementing the Pinnacle system from PAETEC to replace these legacy and home grown systems. At this point we anticipate no significant difficulty in pulling VoIP CDR in.

14 Underlying IP Network

The IP network that supports the VoIP service (and all other data networking) is designed for maximum reliability (99.9% or better), including remaining up for one hour or longer during a power outage so that IP phones may be used for emergency calls. Alternate means of communication are also available should the IP network or VoIP system fail. These include use of cell phones, analog Verizon POTS lines, and fire alarm pull boxes.

14.1 IP address allocation

Globally routable IP v4 addresses are assigned to each phone. In the short term we have adequate capacity and will not use Network Address Translation (NAT). Phones are assigned to a voice-only VLAN and as such will acquire DHCP-assigned addresses from a dedicated pool. In the longer term, IPv6 addresses will be used with soft phones and future hard phones.

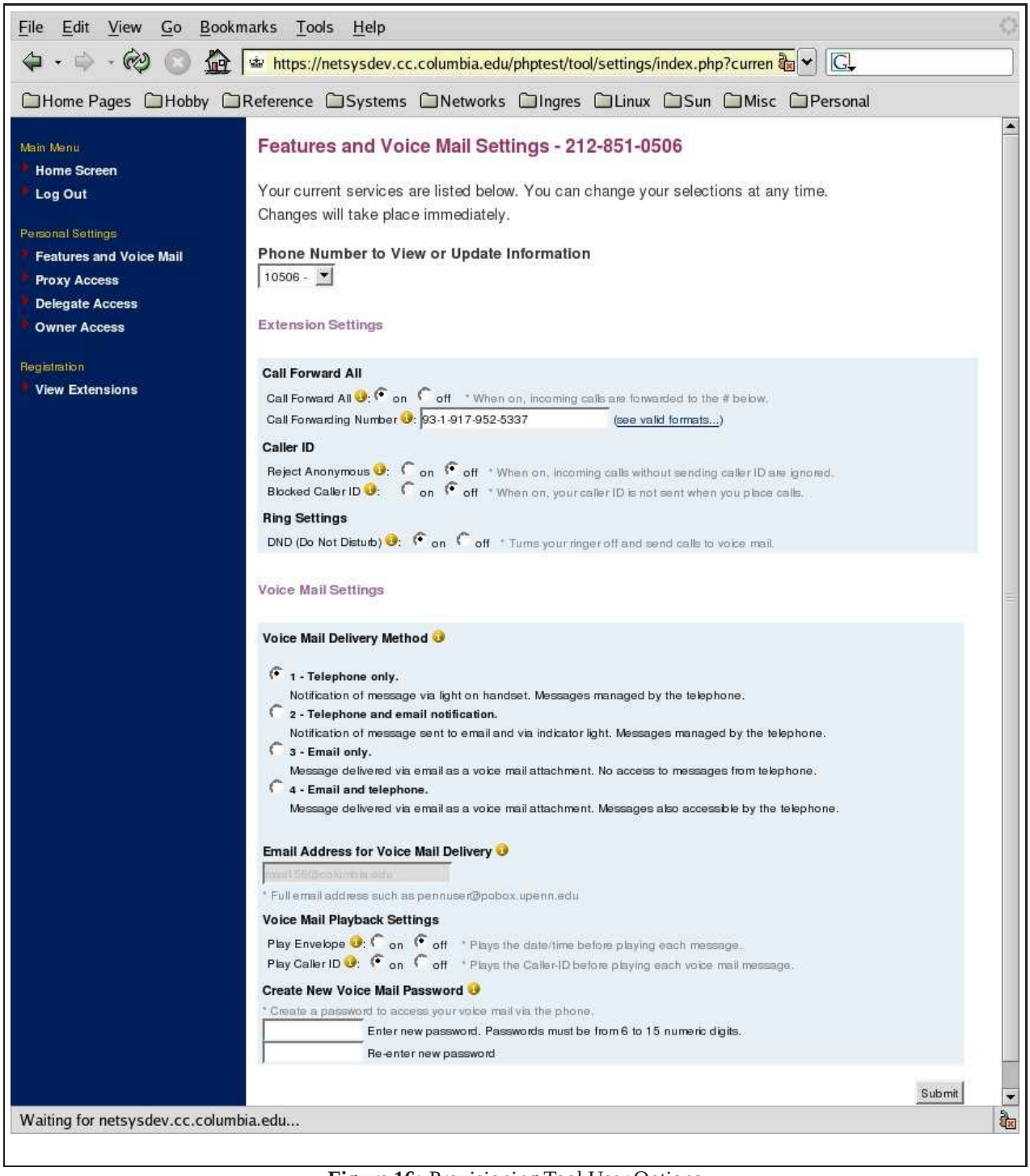

Figure 16: Provisioning Tool User Options

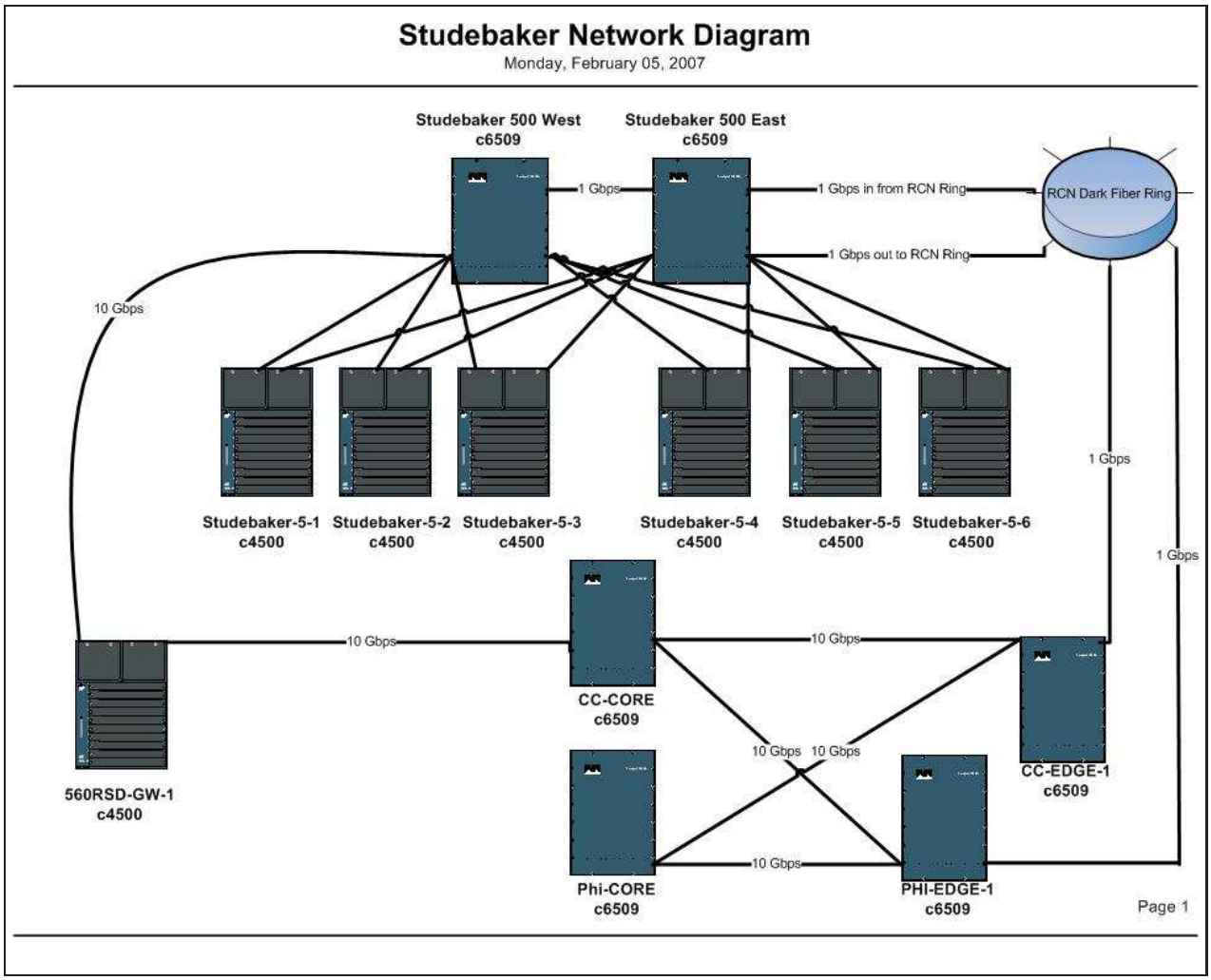

Figure 17: Studebaker building network

14.2 Building access switches

The building access switches used in Studebaker will be Cisco 4500s with Supervisor 2+. VoIP phone ports support IEEE 802.3af Power over Ethernet. CDP and 802.1q are used to select the voice VLAN. The native VLAN is the data VLAN. Each of two floor switches (located on the East and West sides of the building) has two fiber gigabit Ethernet uplinks to the building routers. The uplinks take diverse paths through the building. This diversity protects the switch connectivity against a localized problem such as damage to one of the fiber uplinks caused by construction work.

Each switch has redundant (2N) power supplies. Each power supply is connected to an independent UPS that has sufficient battery capacity for one hour of runtime. Each UPS is powered from an independent 20A power circuit from alternate feeders.

14.3 Distribution routers

Each switch uplinks to two distribution routers on diverse fiber paths. These routers use HSRP to implement redundancy for each VLAN. Like the switches, they have 2N redundant power supplies, dual UPSes and dual power feeds. Distribution routers should be on generator-backed service if at all possible. At this point, sufficient funding to do so in Studebaker has not been allocated.

14.4 Core routers

Two Cisco 6500/Sup720 core routers form a dual-star topology for 10 gigabit Ethernet uplinks from distribution routers. The core routers are physically separated (in Computer Center and Philosophy) and are protected by UPS and generator backup.

14.5 Edge routers

Two Cisco 6500/Sup720 edge routers on the NYC metro fiber ring are located at 111 Eighth Avenue and 32 Avenue of the Americas. Both facilities are robust colocation facilities with UPS and generator-backed power. The ring is implemented with 10 gigabit Ethernet using long-haul XENPAKs. There are no additional active electronics (e.g. repeaters) on the ring. Commodity Internet access via Broadwing is provided at 111 Eighth. Research Internet (NYSERNet, Internet2, NLR, etc.) access is at 32 AofA. NYSERNet has a backup arrangement with Broadwing which will reroute via the NYSERNet statewide network to Buffalo where there is a secondary Broadwing connection. Additional direct peering with ITSPs is being investigated with connectivity likely to happen at one or both of 111 Eighth and 32 AofA.

A third backup edge router is currently installed at Nevis Labs in Irvington, NY. This router has a DS-3 connection to Qwest which is back-hauled via the Qwest protected SONET network to their Boston Internet POP. Connectivity from the Morningside campus to Qwest is via a fast Ethernet Verizon Optical Network (VON) circuit to Lamont-Doherty Earth Observatory and redundant trans-Hudson private 100 Mbps microwave links (one licensed at 18 GHz and one unlicensed U-NII at 5.8 GHz). Pairs of primary and backup routers connect these links.

14.6 Redundant diverse outside fiber plant

A fiber ring connecting Studebaker to the main campus is being constructed. This consists of extension of private fiber that has been built from campus, across Broadway through Barnard College, Union Theological Seminary, Manhattan School of Music, International House and Columbia residential properties extending up Riverside Drive to 125th Street. This will be extended up 12th Avenue and across 132nd Street and into Studebaker from the North side. At 560 Riverside Drive there is an additional 420 Mbps microwave link to the S.W. Mudd building on campus.

Columbia also has a leased dark fiber ring from RCN which is routed up Broadway and down Amsterdam Avenue. This ring will be connected into via a lateral from an ECS manhole on Broadway across 131st Street and into Studebaker from the South side. The connection into the RCN ring and extension of the private fiber build from 560 Riverside Drive (at 125th Street) will close the Manhattanville ring. Within Studebaker, diverse routers on opposite sides of the 5th floor will connect to the North and South entering street laterals.

14.7 Voice VLAN Isolation

Using CDP, Polycom phones cooperate with Cisco switches to use 802.1q trunking to tag voice traffic for a voice VLAN and data traffic to the native VLAN. Every location supporting VoIP has been configured with voice VLANs. These VLANs have public IP addresses to enable direct Internet routing of calls without the need for middle boxes such as Session Border Controllers. To protect these VLANs against SIP signaling spoofing and attacks against other network services on the phone, ACLs restrict:

- UDP and TCP SIP signaling on ports 5060 and 5061 between the voice VLANs and our SIP proxies
- UDP traffic inbound to the well-known Polycom RTP port range of 2222 - 2268
- DHCP between the phones and DHCP servers
- HTTP/HTTPS between the phones and provisioning servers

See the QoS description below for further rate limiting and prioritization that is performed.

14.8 QoS

Quality of service for VoIP calls is assured through the use of DiffServ marking of IP packets throughout the campus network infrastructure and use of priority queues and rate limiters to protect VoIP traffic from excessive bandwidth consumption related to DoS attacks.

DSCP & CoS values

DSCP	CoS	Description	Queuing Algorithm
0	0	Best Effort (BE)	Low priority – WRED
8	1	Scavenger (CS1)	Low priority – WRED
18	2	General services (AF21)	Medium priority – WRED
26	3	Critical services (AF31)	Medium priority – WRED
34	4	VoIP call control, network control	High priority (AF41) – tail drop
56	5	VoIP media (EF)	Priority queue – tail drop

Ethernet switch QoS implementation

Ethernet switches are configured to ignore (rewrite) DSCP/CoS values received on host ports. Specific Cisco Catalyst switch platforms have different capabilities and DSCP/CoS will be addressed as follows:

5000 series QoS is not supported on these obsolete switches. Marking will occur at ingress ports of 6500 routers.

4000 CatOS Voice VLANs are supported. CoS values are trusted. Unmarked traffic can be marked with a default CoS. Interfaces have 2 transmit queues with 1 tail-drop threshold each (2q1t). No input scheduling is available.

4000 IOS Ingress ports may be marked and classified using DSCP or CoS. Per-port trust levels can be configured. The “trusted boundary” feature is used to identify and trust traffic from VoIP phones. Interfaces have 4 transmit queues, one with priority, each with one tail-drop threshold (1p3q1t) and no input scheduling.

Polycom phones Set DSCP and CoS for voice traffic; CoS only for non-voice traffic.

14.9 DHCP, DNS, TFTP, HTTP

XXX

15 Interoperability and Transition Issues

A number of interoperability and transitions issues exist among the VoIP systems and the Rolm PBX.

15.1 Calling and called name display

VoIP to VoIP

Between SIP UAs (a pure VoIP call), caller ID name and number is signaled with the From and Remote-Party-ID headers in the INVITE method. Called number is signaled in the To header. Furthermore, Polycom phones will display the called party name if it is returned as a Remote-Party-ID header in the 180 or 183 response to an INVITE.

VoIP to PBX

Between SIP UAs and the media gateways attached to the Rolm PBX, these headers are translated to/from ISDN Q.931 signaling of both the caller ID (CLID) and calling name (CNAM) information elements. Called name is not signaled back via the 183 Session Progress response. This is worked around by "dipping" the rolm_subscribers table in OpenSER and using this information to insert a Remote-Party-ID header in the 183 response to an INVITE.

PBX to VoIP

CNAM and CLID are converted by the media gateway into a From header. How to send Called Party Name back to the Rolm PBX is currently being studied. Given that this information element is not signaled across the Q.931 interface, it is unlikely that this will work as there does not appear to be a technical means of implementing a database "dip" capability.

VoIP to PSTN

In at least one of our peering agreements (Qwest), the carrier does pass the SIP From and/or Remote-Party-ID display name into the Q.931 CNAM Information Element. However, most PSTN carriers ignore this signaling and dip a static database instead so our ability to pass calling name is limited largely by the legacy PSTN.

PSTN to VoIP

DID is still in the process of being tested with PAETEC. We will accept, and the Polycom phones will display, From/Remote-Party ID display names if provided by the PSTN carriers.

15.2 Phonemail/Voicemail forwarding

Forwarding of received Rolm Phonemail messages to VoIP Voicemail is not implemented nor is the reverse direction. We are considering converting all Phonemail to use VoIP Voicemail. A requirement to implement this is support of translation of the Redirecting Number information element to the Diversion [33, 34] or Referred-By [16] header. This needs to be supplied by the HiCom 300 in the Q.931 signaling at which point the Cisco media gateway should translate this to a Diversion header. We have so far been unable to get Diversion to work. This feature may not work on the Rolm PBX.

However, we have discovered that the Intel PIMG product will work as a special case media gateway for PhoneMail. The PIMG connects to the Rolm system via RolmLink Interfaces (RLI – proprietary digital interface used between RolmPhones and the PBX) and routes calls via SIP, providing Diversion (Referred-By???) headers and translating SIP NOTIFY into turning on and off the MWI. A PIMG has been acquired and is being tested as a potential means of replacing PhoneMail with SIP voicemail. At least one other Rolm customer today uses the PIMG with Asterisk voicemail.

15.3 Hairpin forwarding

The use case that triggers this issue is that of a Rolm phone being set to forward to a VoIP phone. When the Rolm extension is called from another VoIP phone, a hairpin call is created in which the calling VoIP phone should either get a SIP REFER from the media gateway, or a new call leg should be initiated from the PBX back toward the forwarding target VoIP phone. The REFER is preferred as it does not tie up two trunks between the PBX and media gateway and is the logically correct method. In working with Siemens and Cisco we have determined that this is likely a a Cisco media gateway interoperability issue. This case is currently open with the Cisco TAC and engineering groups to resolve the inability of the media gateway to properly translate a QSIG Redirect Information Element into a SIP REFER. An offered workaround, “2 B” which ties up two trunks is not implemented on the Siemens side.

In the event that a satisfactory workaround is not available, we will have to document a two-step forwarding procedure that will both set a forwarding destination on the Rolm PBX and in the OpenSER proxy. In this scenario a VoIP call will never hairpin through the Rolm. However, this will require coordination with two places that both have to have forwarding set.

15.4 Use of Expired Internet Draft RFCs

Many proposed Internet standards (RFCs) are promulgated as draft proposals some of which never make it through the IETF vetting process. These drafts frequently define key necessary features for a PBX replacement SIP implementation. Vendors of SIP products frequently implement these drafts with the expectation that they will eventually be standardized. In many cases, support for drafts that fail to be standardized is retained to implement key product features and vendors maintain that support despite the failure of the draft to become a standard.

A goal of the CUIT VoIP product is to rely solely on standard SIP components as defined in the applicable RFCs while tracking new developments through the RFC draft process. However, some features we require may never become standardized. We need to be careful on how much we rely on these features and especially be cognizant of the possibility that our vendors will drop support in future code releases.

Examples of key drafts that are widely implemented but have expired include the Remote-Party-ID [18] and Diversion [33] headers.

Remote-Party-ID

Remote-Party-ID is used primarily to provide the Caller ID name and number of a SIP caller and to transition it to the PSTN equivalent. Polycom phones use this header both in INVITEs (Caller ID) and in 18x Ringing responses (Called Party ID). Caller ID can be worked around by using a B2BUA implementation that modifies the From header's display name and URI appropriately. Called Party ID (supplied by the called party's network) can not be implemented in this manner and appears to require the RPID. RPID has been replaced by the P-Asserted-Identity header [15].

Diversion

The Diversion header indicates the original target of a call that has been forwarded. This capability is required, for example, to notify the called party that the call they are receiving was forwarded by an intermediary and is not directly from the apparent calling party (the original caller). Our Rolm PBX implements this feature ("FWD" display on the phone when a call is ringing). Diversion is also used for applications like forwarding to voicemail so that the originally called number is available to the voicemail system. Diversion has been replaced by the Referred-By header [16].

16 Security and Privacy

16.1 Encryption of Signaling and Media

Today, VoIP calls typically cross the network unencrypted. SIP signaling and media are transmitted in the clear and can be fairly easily intercepted. In Phase I of the project, we will live with this risk. Calls will all be on a physically secured wired network within Studebaker and support of remote (road warrior) users will not be offered. In Phase II, we will add support for TLS encryption of SIP signaling as well as media encryption for the RTP media streams [24]. SIP/TLS (SIPS) support is available today in the Polycom phones and OpenSER proxy. Media encryption support from vendors (Polycom phones, Cisco Media Gateways, etc.) is expected in the next year or so. ITSP carriers do not today offer session encryption but this has been identified as a requirement from them going forward.

16.2 Anonymous Calling

Anonymous calling is not something we need or support today within the campus phone network. Using VoIP, it is difficult to make a truly anonymous call: the IP addresses of the endpoints are visible to each other, for example. Even if NAT or a back-to-back user agent (B2BUA) is implemented to anonymize calls, they are still traceable back to the institution.

Anonymous calling across the PSTN is implemented by our ITSPs, although we are trusting them to implement it. The Remote-Party-ID and From headers are adjusted appropriately for anonymous INVITEs but the carrier still has visibility into the SIP Contact header which discloses the SIP user and IP address of the phone.

16.3 Spoofing

Spoofing of Caller ID for SIP calls is possible just like with PSTN calls. There is no method of authenticating caller ID in the PSTN. In SIP, it is possible to implement anti-spoofing measures using, for example, P-Asserted-Identity [15] and future developments that will provide end-to-end identity assertion.

16.4 Theft of Service

Theft of service is accomplished in SIP by learning the SIP user and password for a phone. The SIP user is a publicly known value (the extension number in our implementation). The password is encrypted over the network using WWW digest authentication so can not be captured. However, if the phone itself is compromised or the installation of the password into the phone is snooped, this can be a problem. This is why we are using HTTPS provisioning and AES encryption of the SIP registration file for the phone.

Of course, anybody who has physical access to the phone can use it to place calls. In insecure areas, phones are provisioned with restricted access and require entry of a PSC code to be able to dial toll calls.

16.5 Media Gateway Security

Access to the media gateway, which does not implement SIP authentication, is protected via an Access Control List which blocks SIP signaling traffic (on port 5060) from anywhere other than our proxies. This prevents someone from relaying PSTN calls through our gateways and the Rolm PBX.

16.6 Phone Physical Security

If someone steals a RolmPhone, it doesn't have much value these days. If they steal a Polycom phone, it is possible to:

- Plug it in on the Internet somewhere and place calls billed to the customer of record.
- Erase the phone's configuration, making it valuable for use on other SIP networks.

If the first item happens, we will see in our logs that the phone is no longer at the IP address we expect to be and further will have the IP address it is being used from which can be turned over to law enforcement if necessary. So, we will need to audit our logs of SIP phone registrations to make sure the phones stay where they are expected. With our MAC address polling mechanisms, we can determine the exact on-campus address of a SIP phone.

If the latter item happens, we are out the replacement cost of the phone (just as we are when a computer or any other item is stolen). It is probably not worth the inconvenience to physically lock down $200 desk phones.

16.7 SIP SPAM (SPIT)

Like Email SPAM, and PSTN junk calls, we can expect SIP SPAM to start coming soon. In fact, it's already been named SPAM over Internet Telephony or SPIT [35, 36, 37]. When and if SPIT becomes a problem, we will track the community techniques used to combat it and implement them. Currently, we do not accept spoofed identity (From header) from our own users. Use of P-Asserted-Identity will be a likely next step for Internet callers when we implement Internet call routing.

16.8 Lawful Intercept

The Communications Assistance for Law Enforcement Act of 1994, CALEA [38], requires PSTN providers to enable automated legal wiretaps [39]. A recent FCC modification of the rules [40] has extended the definition of CALEA to include *interconnected* VoIP systems. It is our position based on opinions from Educause, ACE, ACUTA, and others that the newly extended CALEA does not apply to our private VoIP network [41, 42, 43, 44, 45] just as CALEA has never applied to our PBX. Our VoIP systems are connected by ITSPs and TDM trunking carriers who are required to comply with CALEA on their networks.

The new CALEA rules go into effect in May, 2007. Even though the effective date of these rules is imminent, to date no visible action has been taken to define the technical standards required to implement lawful intercept on an IP network. In the event that CALEA compliance is determined to apply to our network, it will likely require software and/or hardware updates to our three border routers where we peer with commodity Internet and research networks (NYSERNet, Internet2, CERN, NLR).

17 Diagnostic Tools

A number of diagnostic tools are used to ensure VoIP quality.

17.1 Network Qualification Testing

When installed and as needed, end-user network ports are tested with the IPERF tool to assure a minimum of 90 Mbps end-to-end to an on-campus testing endpoint with zero packet loss and consistent latency < 1 ms. In addition to use of IPERF, BWCTL and NDT will be used with test hosts installed in critical building telecommunication rooms to periodically generate test traffic. For more information on these tools see http://e2epi.internet2.edu.

17.2 Network Interface and Performance Monitoring

SNMP polling of all network interfaces (switch and router links) is performed at 5 minute intervals. This data is graphed using Cricket and summarized over time, providing near-real time as well as historical performance reports. The Intermapper software is also used to monitor the network and to alert the NOC to link down, excess traffic and excess error rates. Errors caused by dirty fiber uplinks and other hardware problems are referred to the Field Services group for repair.

17.3 Switch and Router Log Processing

Daily summarization of syslogs from router and switches is performed and the NOC staff review this data and identify problems (e.g. link and routing protocol flaps), diagnose, and correct them or refer to Field Services for hardware repair. See figure 20.

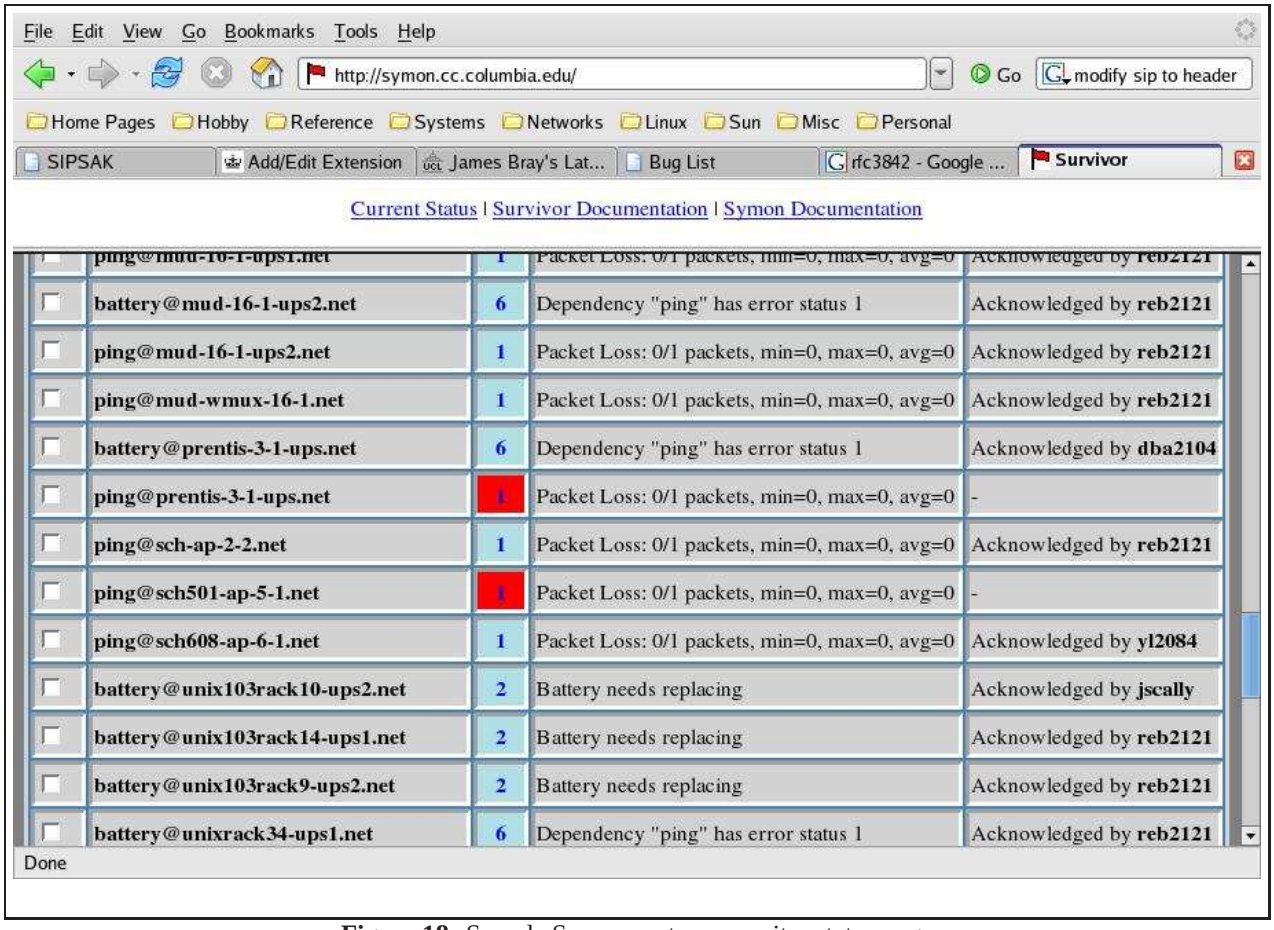

Figure 18: Sample Symon systems monitor status page

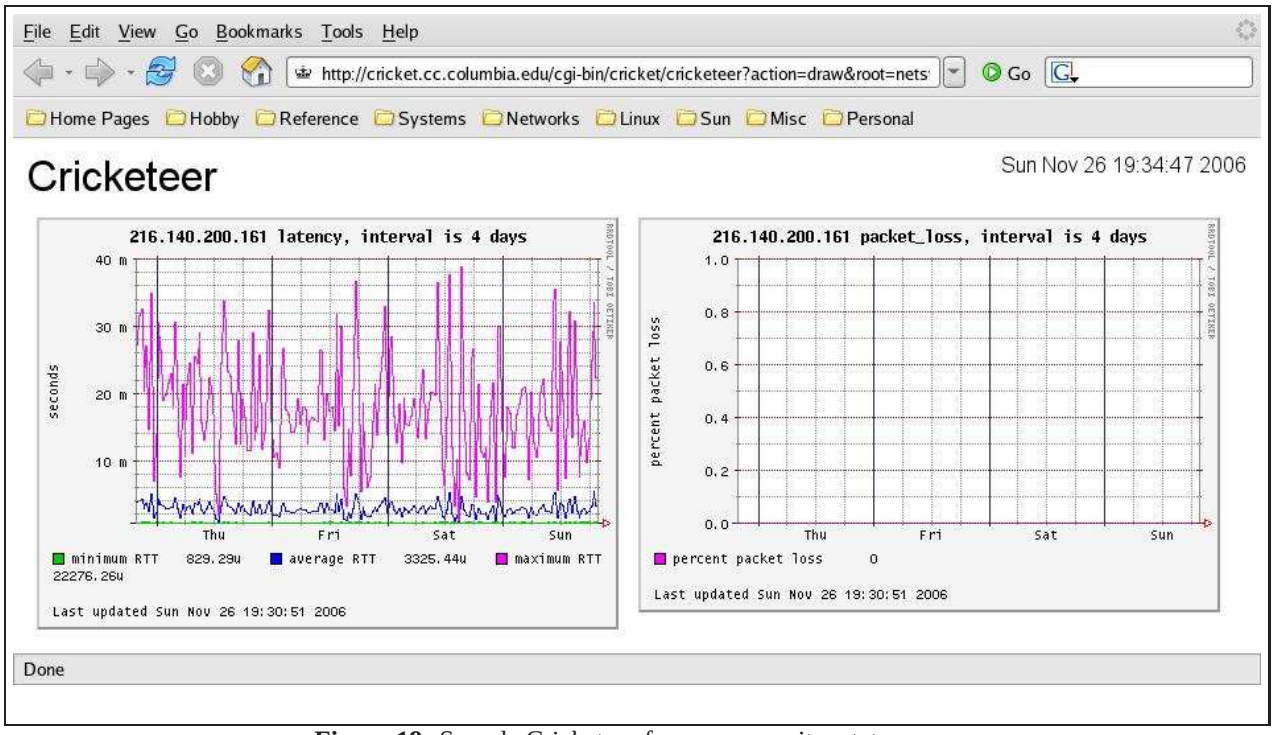

Figure 19: Sample Cricket performance monitor status page

```
mudd-edge-1.net:

  %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet4/8, changed state to down (11 times) (11:24:47)

  %IP-4-DUPADDR: Duplicate address 160.39.61.129 on Vlan961, sourced by 0800.46cc.3111 (95 times) (11:08:20)

  %LINK-3-UPDOWN: Interface GigabitEthernet4/8, changed state to down (11 times) (11:24:48)

  %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet4/8, changed state to up (11 times) (11:25:01)

  %LINK-3-UPDOWN: Interface GigabitEthernet4/8, changed state to up (11 times) (11:25:00)

Summary of /bianaoh/log/cisco/switch/switch:

butl-2-1.net:

  %SYS-2-TEMP_HIGHOK:Temp high okay (2 times) (10:22:19)

  %SYS-2-TEMP_HIGHFAIL:Temp high failure (2 times) (06:51:39)
```

Figure 20: Sample Cisco log summary

17.4 SIP Activity Logging

All SIP call activity through the proxies is regularly syslogged and can be referenced to debug reported problems. Figure 21 shows an example of a missed call which goes to voicemail. Besides the usual syslog timestamp, hostname and process name and ID headers, all openSER logs have been configured to include the SIP Call-ID which uniquely identifies a call.

17.5 Phone Logs

Polycom phones upload their log files to the provisioning server periodically and on demand by pressing the four arrow keys simultaneously. These logs are analyzed when needed for detailed debugging. See figure 22 for an example.

17.6 SIP Packet Capture

Using tools like tcpdump, ngrep and wireshark, SIP signaling (and media) packet traces can be captured in real time when reproducing and debugging a problem. Wireshark (formerly known as Ethereal) does an excellent job of decoding all SIP protocol headers and includes the ability to display a session protocol trace in a time line fashion as well (figure 23). The Cisco switches in our network all support "spanning" a port to enable packet capture of a SIP dialog.

17.7 Measuring Mean Opinion Score (MOS)

It is difficult to measure MOS directly as it is a subjective measurement. However, RTCP[46] and RTCP XR[47] can be used to track measured packet loss, jitter and delay. We are investigating methods of collecting and analyzing RTCP data from calls to detect performance problems.

18 Operational Support Plan

The support plan for this CUIT service is similar to support plans for other CUIT services.

18.1 Service Request Channel

1. Standard-urgency issues should be reported through: CUIT Online Support Center - http://www.askcuit.columbia.edu/
2. Non-urgent questions or comments can be sent to: askcuitcolumbia.edu
3. Urgent issues should be reported to CUIT Client Service Desk. 212-854-1919 (Su 3p-11p, M-Th 8a-11p, Fr 8a-7p, Sa 10a-6p) Please Note: customers experiencing problems with their own VoIP phone device will need to use another phone to call.

```
Nov  6 17:03:02 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[0] INVITE r-uri <sip:10508@siptest.columbia.edu;user=phone>
 from <sip:10501@siptest.columbia.edu> to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:02 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[2] INVITE r-uri <sip:10508@128.59.37.206> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:02 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 sip:10508@siptest.columbia.edu;user=phone is a voicemail user
Nov  6 17:03:02 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[2] ruri 10508 firstname Alan lastname Crosswell
Nov  6 17:03:02 jello /usr/sbin/openser[1634]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 onreply_route[2] INVITE  r-uri <<null>> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone> status 100 Trying
Nov  6 17:03:02 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 onreply_route[2] INVITE  r-uri <<null>> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone> status 180 Ringing
Nov  6 17:03:02 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 onreply_route[2] stuffing Alan Crosswell as callee name
Nov  6 17:03:02 jello /usr/sbin/openser[1637]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[0] PRACK r-uri <sip:10508@128.59.37.206> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:02 jello /usr/sbin/openser[1637]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[1] PRACK r-uri <sip:10508@128.59.37.206> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:02 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[0] PRACK r-uri <sip:10508@128.59.37.206> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:02 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[1] PRACK r-uri <sip:10508@128.59.37.206> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:06 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[0] CANCEL r-uri <sip:10508@siptest.columbia.edu;user=phone>
 from <sip:10501@siptest.columbia.edu> to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:06 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 leave route[0]: CANCEL r-uri <sip:10508@siptest.columbia.edu;user=phone>
 from <sip:10501@siptest.columbia.edu> to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:07 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 onreply_route[2] INVITE  r-uri <<null>> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone> status 487 Request Cancelled
Nov  6 17:03:07 jello /usr/sbin/openser[1640]: ACC: call missed: method=INVITE,
 i-uri=sip:10508@siptest.columbia.edu;user=phone, o-uri=sip:10508@128.59.37.206,
 call_id=76107cb3-7e8aef1c-4c108755@128.59.115.115,
 from="Alan Crosswell" <sip:10501@siptest.columbia.edu>;tag=A25DA65E-EC89FD37,
 code=487 Request Cancelled
Nov  6 17:03:07 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 failure_route[1] INVITE r-uri <sip:10508@128.59.37.206>
 from <sip:10501@siptest.columbia.edu> to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:07 jello /usr/sbin/openser[1640]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 redirection to voicemail from sip:10501@siptest.columbia.edu
 for sip:10508@siptest.columbia.edu;user=phone
Nov  6 17:03:07 jello /usr/sbin/openser[1634]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[0] ACK r-uri <sip:10508@128.59.37.206> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone>
Nov  6 17:03:07 jello /usr/sbin/openser[1634]: 76107cb3-7e8aef1c-4c108755@128.59.115.115:
 route[1] ACK r-uri <sip:10508@128.59.37.206> from <sip:10501@siptest.columbia.edu>
 to <sip:10508@siptest.columbia.edu;user=phone>
```

Figure 21: Sample of OpenSER Activity Log

```
0419141145|app1 |*|02|Initial log entry. Current logging level 4
0419141145|app1 |4|02|mb.main.home parameter is empty
0419141145|mb   |*|02|Initial log entry. Current logging level 4
0419141154|so   |*|02|[SoNcasC]: App-Ctx (Fitzgerald Woolcott) [0-43417@sip.columbia.edu]
0419141203|slog |*|02|Initial log entry. Current logging level 4
0419141203|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep2.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141203|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep2.bmp' is not present
0419141203|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep2.bmp, errno 0x388002.
0419141203|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep3.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141203|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep3.bmp' is not present
0419141203|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep3.bmp, errno 0x388002.
0419141203|cfg  |4|02|Mgr|Poll in 46179 seconds
0419141203|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep4.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141203|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep4.bmp' is not present
0419141203|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep4.bmp, errno 0x388002.
0419141203|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep5.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141203|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep5.bmp' is not present
0419141203|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep5.bmp, errno 0x388002.
0419141238|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep6.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141238|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep6.bmp' is not present
0419141238|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep6.bmp, errno 0x388002.
0419141238|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep7.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141238|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep7.bmp' is not present
0419141238|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep7.bmp, errno 0x388002.
0419141238|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep8.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141238|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep8.bmp' is not present
0419141238|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep8.bmp, errno 0x388002.
0419141238|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep9.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141238|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep9.bmp' is not present
0419141238|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep9.bmp, errno 0x388002.
0419141238|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep10.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141238|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep10.bmp' is not present
0419141238|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep10.bmp, errno 0x388002.
0419141238|copy |4|02|Download of '~alan/poly/initial/CtxActiveHDStep11.bmp' FAILED on attempt 1 (addr 1 of 1)
0419141238|copy |4|02|Server 'www.columbia.edu' said '~alan/poly/initial/CtxActiveHDStep11.bmp' is not present
0419141238|res  |4|02|[ResFinderC]: Failed to download file CtxActiveHDStep11.bmp, errno 0x388002.
0419141239|cfg  |4|02|Mgr|Poll in 46143 seconds
0419141737|copy |4|02|Upload of 'phonefiles/0004f204fe66-app.log' FAILED on attempt 1 (addr 1 of 1)
0419141743|copy |4|02|Upload of 'phonefiles/0004f204fe66-app.log' succeeded on attempt 2 (addr 1 of 1)
0419142250|copy |4|02|Upload of 'phonefiles/0004f204fe66-app.log' FAILED on attempt 1 (addr 1 of 1)
0419142257|copy |4|02|Upload of 'phonefiles/0004f204fe66-app.log' succeeded on attempt 2 (addr 1 of 1)
0420030142|cfg  |4|02|Mgr|CfgMgr check for changes
0420030142|cfg  |4|02|Mgr|Poll in 86400 seconds
0420105223|net  |4|02|rtosNetwork: netwTask() - Can't find associated CCB!
0420141235|cfg  |4|02|Mgr|Poll in 46147 seconds
0421030143|cfg  |4|02|Mgr|CfgMgr check for changes
0421030143|cfg  |4|02|Mgr|Poll in 86399 seconds
0421141236|cfg  |4|02|Mgr|Poll in 46146 seconds
```

Figure 22: Sample of Uploaded Polycom Phone Log

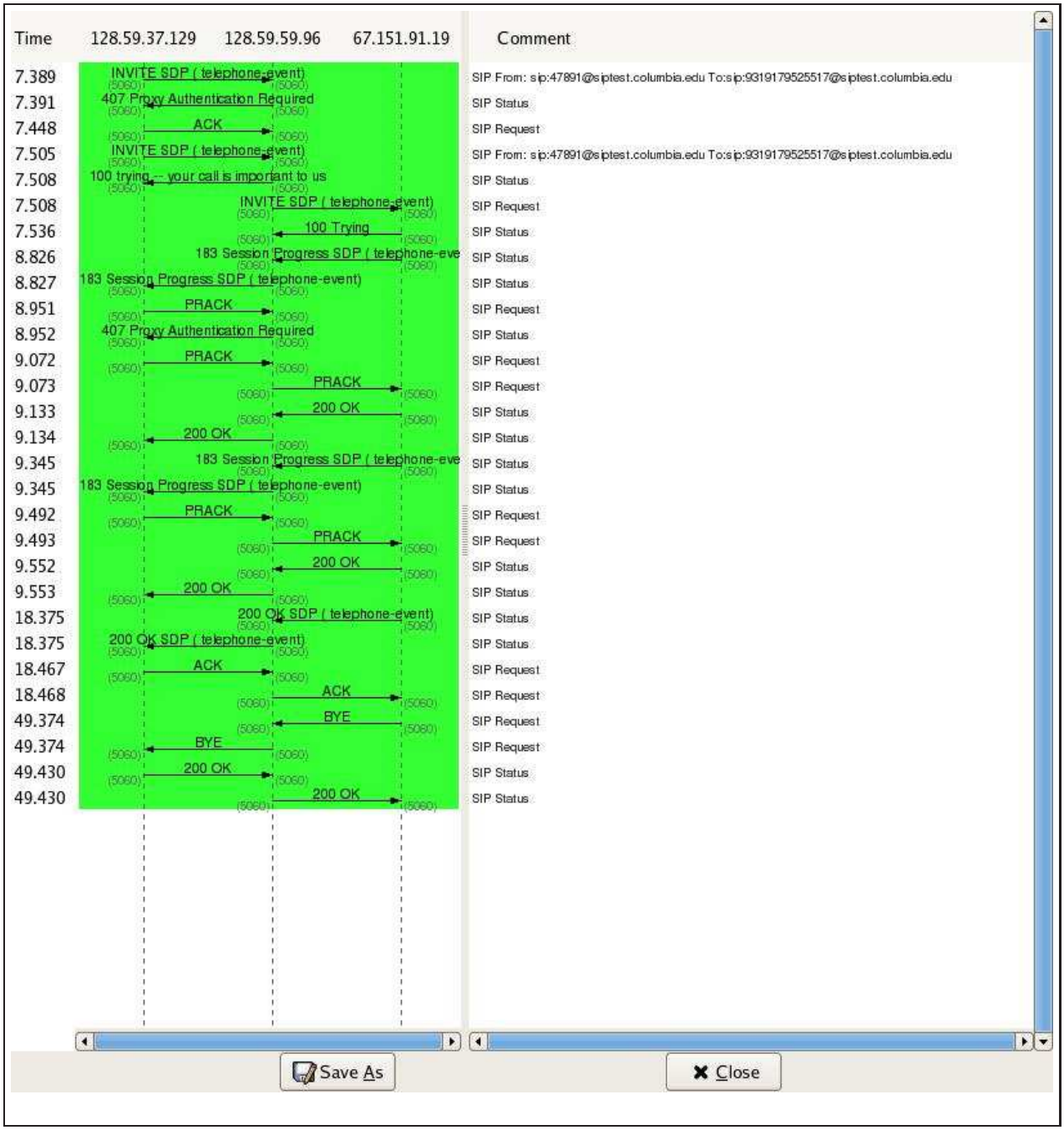

Figure 23: Sample of Wireshark Call Flow Graph

18.2 Issue Reporting and Escalation Procedure

Remedy Escalation

Upon processing a support request, CUIT Service Desk group will assign the Remedy ticket to the VoIP Support queue.

Ownership Tree:

1st Level Support: Client Service Desk group

2nd Level Support: CTS Design group

3rd Level Support: Network Operations Center group

Ownership Details:

1. The VoIP Support queue is jointly managed by the CTS Design and Network Operations Center groups.
2. Initial ownership within this queue is provided by the CTS Design group.
3. Any issues that the CTS Design group cannot solve in *time period* are escalated to the NOC support staff.
4. If the NOC staff cannot resolve the user's issue in *time period* the NOC staff retains ownership of the issue and consults with Network Software Systems, Polycom, OpenSER and/or Asterisk support until the issue is resolved.

19 Regression Test Plan

A regression test suite has been developed that is based on original work both by CUIT and the Polycom Vendor Interoperability Program (VIP) test suite. The test plan is available on the NOC wiki.[2]

This test suite continues to evolve as new features are implemented.

19.1 Testing endpoints

Table 3 summarizes the endpoints referenced in this plan.

19.2 Basic Call Tests

Polycom 430 dial plan using line key

This tests the ability to place basic calls and exercise the Polycom phone local dial plan as well as the OpenSER dial plan for call routing.

[2] https://www1.columbia.edu/sec/acis/netsys/manual/noc/voip/testScenarios.html

Key	Type/Location	Test Number	Comments
		5-digit internal extensions	
A	Polycom 430	43418	
B	Polycom 601	43417	
C	4xxxx	41288	Rolm Phonemail
D	3xxxx	30000	Rolm Phonemail
E	1xxxx	12298	Rolm analog line w/Fax
		10-digit local NYC numbers	
F	212 information	93-1-212-555-1212	
G	718 information	93-1-718-555-1212	
H	347 information	93-1-347-555-1212	
I	646 information	93-1-646-555-1212	
J	917 information	93-1-917-555-1212	
		10 digit long distance	
K	regional fax	93-1-215-937-6382	
L	national fax	93-1-702-946-8312	
		cellular	
M	NOC cell phone	93-1-917-670-4007	
		International	
N	Paris, France	93-011-33-144413202	
O	Australia (Bureau of Meteorology)	93-011-61-396694916	
P	England (US Embassy)	93-011-44-2074999000	Do not select the item that asks to connect you to the US Marine Guard
		Inter-campus tie lines	
Q	Interchurch Center fax	80-3399	
R	Lamont-Doherty fax	95-8736	
S	Nevis Labs Security	72-2870	
T	Teachers College fax	94-3222	
U	Reid Hall fax	53-3202	
V	CUMC fax	51-23914	
		HiPath 4000/Agile	
W	CUIT Help Desk	41919	
		Personal Security Codes	
X		97+0173197663 + local tel. number	You can try any local number e.g. 917-670-4007
		Special Digits	
Y	CU Public Safety	99	Should translate to Rolm ext 45555
Z	CU operators	90	Should translate to ext 41754
		911-Emergency	
AA	911	911	Remember to state that this is a test call and not an emergency!
AB	911	93-911	
		ATT-Operator	
AC	ATT Operator	93+00	This connects you to a ATT operator for credit card calls

Table 3: Regression testing endpoints

1. On phone A, press the line key and get a dialtone.
2. Dial each of B-E, K-W and ensure that you are able to complete the call. Expect to hear secondary dialtone after entering an outside or tie line prefix, ring indication, and have the call answered.

Polycom 430 dial plan using Dial soft key

This is essentially the same as the prior test but uses the alternate method of entering the target number on the phone and then hitting "Dial" to cause the call to be place.

1. On phone A, press the line key and get a dialtone.
2. Dial each of B-E, K-W and ensure that you are able to complete the call. Expect to hear ring indication, and have the call answered.

Polycom 601 dial plan using line key

This is a repeat of the previous test for the 601 model. While these phones are logically equivalent, the hardware differs (the 601 is the older processor model) and we have seen divergent behavior between the two models from time to time.

1. On phone B, press the line key and get a dialtone.
2. Dial each of A, C-E, K-W and ensure that you are able to complete the call. Expect to hear secondary dialtone after entering an outside or tie line prefix, ring indication, and have the call answered.

Polycom 601 dial plan using Dial soft key

This is essentially the same as the prior test but uses the alternate method of entering the target number on the phone and then hitting "Dial" to cause the call to be place.

1. On phone B, press the line key and get a dialtone.
2. Dial each of A, C-E, K-W and ensure that you are able to complete the call. Expect to hear ring indication, and have the call answered.

19.3 Polycom Speaker Test

Test speaker phone calls between two Polycoms

1. On phone A, press line key, speaker key or New Call soft key to obtain dialtone.
2. Dial phone B.
3. On phone B, answer the call using the handset and communicate with phone A user.
4. On phone B, press the speaker key.

5. Confirm that phone A and B users can communicate clearly using the full-duplex speaker phones.
6. XXX - in NOC test plan this also tests Hold/Resume which should move elsewhere

19.4 Polycom Headset Test

Test headset phones calls between two Polycoms

1. Connect a headset to phone B.
2. On phone B, press headset Key and call phone A.
3. Confirm phone A and B users can communicate.

19.5 Polycom Mute Test

Test mute key on Polycom 430 and 601.

1. On phone A, dial phone B.
2. On phone A, press the Mute soft key.
3. Confirm that phone A's Mute light is on and that the line display shows the mute icon.
4. Confirm that phone B cannot hear person on phone A.
5. Press the Mute key again and Confirm that phone B can hear phone A.

19.6 Polycom Hold Test

Test hold soft key between Polycom phones

This tests the hold implementation which uses re-INVITEs with the RFC3264 method.

1. On phone A, dial phone B.
2. After phone B answers, press the Hold soft key.
3. Confirm that phone A's call to phone B is placed on hold.
4. Confirm that phone A's display shows the Resume soft key.
5. Press Resume and confirm call is taken off hold.
6. XXX - need to test hold behavior across POTS and SIP trunks too

19.7 Polycom Redial Test

Test redial key on Polycom phones

1. On phone A, dial phone B
2. Hang up both phones.
3. On phone A, press the Redial button.
4. Confirm that phone A redials the call to phone B.

19.8 Polycom Call Return Test

Test call return on Polycom 430 and 601 phones

1. Place several incoming calls to each of phone A and B from on-campus, and off-campus phones.
2. Using the phone A arrow keys, select each of the Received or Missed calls list.
3. Move to each of the on-campus 5-digit and off-campus 10-digit numbers and press the Dial soft key.
4. Confirm that the call was placed (no error tone or misrouting).

19.9 Polycom Message Waiting Indicator Test

Leave a message

1. On phone A, dial phone B and let it ring until forwarded to voicemail.
2. Leave a message and hang up.
3. Confirm that phone B's message waiting indicators are on. The indicators include a red blinking light, envelope icon next to the line key on the LCD display, and broken dialtone when picking up phone B.

Retrieve message directly from phone B

1. On phone B, press the Messages button.
2. Confirm that the number of new and old messages displayed is accurate.
3. Press Connect to dial voicemail access.
4. Upon connecting to voicemail, log in, listen to the message and delete it.
5. Confirm that the number of new and old messages reported by voicemail is accurate.
6. Confirm that the message waiting indicators have cleared.

Retrieve message via remote voicemail access

1. Repeat above to leave a new message for phone B.
2. On phone A, dial voicemail access (*86 or 4-8600).
3. Login to voicemail with phone B's extension number and password.
4. Confirm that the number of new and old messages reported by the voicemail is accurate.
5. Listen to the message and delete it.
6. Confirm that the message waiting indicators have cleared on phone B.

19.10 DTMF passthru tests

Test Asterisk voicemail DTMF

1. On phone B, connect to voicemail.
2. Navigate the menus via DTMF.
3. Confirm that DTMF navigation worked.

Test domestic long distance DTMF

1. On phone B, call 1-800-FANDANGO (1-800-326-3264).
2. XXX – add 1-800-FANDANGO and 4-0494 table
3. Navigate the menus via DTMF.
4. Confirm that DTMF navigation worked.

Test international DTMF

1. On phone B, dial phone P.
2. Navigate the menus via DTMF. Note: Do not select the item that connects you to the US Marine guard.
3. Confirm that DTMF navigation worked.

19.11 Call Waiting & Multiple Lines per Registration

Call Waiting test

1. On phone B, dial 4-0494 and remain on the line.
2. On phone A dial phone B.
3. Confirm that phone B hears a call waiting tone and indicates a second incoming call.
4. Place the current call on hold.
5. Pick up the incoming call by pressing the Answer soft key.

Multiple Lines per Registration test

1. XXX - reconfigure phone B to have multiple registrations per line.
2. On phone B, dial 4-0494 and remain on the line.
3. On phone A dial phone B.
4. Confirm that phone B hears a call waiting tone and indicates a second incoming call by illuminating the second line indicator.
5. Place the current call on hold.
6. Pick up the incoming call by pressing the second line key.

19.12 Conference Calling

3-way conference test

1. On phone B, dial 4-0494 and remain on the line.
2. On phone B, press the Cnfrnc soft key.
3. On phone B, dial phone A.
4. Answer phone A.
5. Press the Cnfrnc soft key.
6. Confirm that all three parties can hear each other.

XXX - N-way conference test

This feature is not yet implemented.

1. On phone B, dial 4-0494 and remain on the line.
2. On phone B, press the Cnfrnc soft key.
3. On phone B, dial phone A.
4. Answer phone A.
5. Press the Cnfrnc soft key.
6. Confirm that all three parties can hear each other.
7. On phone B, press the Cnfrnc soft key.
8. On phone B, dial phone XXX.
9. Answer phone XXX.
10. Press the Cnfrnc soft key.
11. Confirm that all four parties can hear each other.

19.13 Restricted Calling

On-campus Only Restriction test

1. Use Provisioning Tool to set phone A's extension to be restricted to on-campus calling.
2. Using phone A, call each of phones B,E,K–N and 1-800-FANDANGO
3. Confirm that only phone B's call went through.
4. Confirm that other calls resulted in an appropriate error message.

Inter-campus Only Restriction test

1. Use Provisioning Tool to set phone A's extension to be restricted to inter-campus calling.
2. Using phone A, call each of phones B,E,K–N and 1-800-FANDANGO
3. Confirm that only phone E's call went through.
4. Confirm that other calls resulted in an appropriate error message.

Local Only Restriction test

1. Use Provisioning Tool to set phone A's extension to be restricted to local calling.
2. Using phone A, call each of phones B,E,K–N and 1-800-FANDANGO
3. Confirm that only phone F's call went through.
4. Confirm that other calls resulted in an appropriate error message.

Long Distance Only Restriction test

1. Use Provisioning Tool to set phone A's extension to be restricted to LD calling.
2. Using phone A, call each of phones B,E,K–N and 1-800-FANDANGO
3. Confirm that only phone K's call went through.
4. Confirm that other calls resulted in an appropriate error message.

International Only Restriction test

1. Use Provisioning Tool to set phone A's extension to be restricted to international calling.
2. Using phone A, call each of phones B,E,K–N and 1-800-FANDANGO
3. Confirm that only phone N's call went through.
4. Confirm that other calls resulted in an appropriate error message.

Toll Free Restriction test

1. Use Provisioning Tool to set phone A's extension to be restricted to toll free calling.
2. Using phone A, call each of phones B,E,K–N and 1-800-FANDANGO
3. Confirm that only the 1-800 call went through.
4. Confirm that other calls resulted in an appropriate error message.

Personal Security Code test

1. Use Provisioning Tool to set phone A's extension to be restricted to on campus calling.
2. Using phone A, call each of phones B,E,K–N and 1-800-FANDANGO, using 97+PSC where appropriate.
3. Confirm that all calls went through when PSC code is used.

19.14 Distinctive Ringing

Internal vs. External Distinctive Ring test

1. On phone A, dial phone B.
2. Confirm that phone B rings with a single (internal) ring.
3. On phone M, dial phone B.
4. Confirm that phone B rings with a double (external) ring.
5. On phone XXX (4-0494), dial phone B.
6. Confirm that phone B rings with a single (internal) ring.

Distinctive Ringing Customization

This feature is not yet implemented.

1. Use the User Provisioning Tool to configure non-default internal and external ring tones for phone B.
2. On phone A, dial phone B.
3. Confirm that phone B rings with the configured internal ring.
4. On phone M, dial phone B.
5. Confirm that phone B rings with the configured external ring.

19.15 Do Not Disturb

Do Not Disturb test

1. On phone B, press the Do Not Disturb button.
2. Confirm the DND visual indication is present on the phone display.
3. On phone A, dial phone B.
4. Confirm that the call does not ring and is forwarded to voicemail.
5. On phone B, press DND again to return to normal operation.

Reject Incoming call test

1. On phone B, dial phone A.
2. While phone A is ringing, press the Reject soft key.
3. Confirm that phone B is connected to phone A's voicemail.

19.16 Anonymous Calling

Anonymous Call Reject test

1. Use the Provisioning Tool to select anonymous call reject for phone B.
2. On phone M, dial *67+ phone B's number.
3. Confirm that phone M's call XXX rings? goes to error recording? XXX

Per-call Restrict test

1. Have phone A dial phone M.
2. Confirm that phone M receives caller ID for phone A.
3. Have phone A dial *67+ phone M's number.
4. Confirm that phone M receives caller ID restrict message for the incoming call.

All-call Restrict test

1. Use the Provisioning Tool to configure phone A for All-call restrict.
2. Have phone A dial phone M.
3. Confirm that phone M receives caller ID restrict message for the incoming call.

Per-call Unrestrict test

1. Use the Provisioning Tool to configure phone A for All-call restrict.
2. Have phone A dial *82 + the number of phone M.
3. Confirm that phone M receives caller ID for the incoming call.

Voicemail

1.
2.
3.
4.
5.

19.17 SIP trunking tests

These tests will force calls to route via specific SIP trunking providers.

1.
2.
3.
4.
5.

20 Disaster Recovery Plan

The following disaster recovery plan is broken down by anticipated types of disaster and the mitigation steps that will be taken.

20.1 General mitigating actions

In the event of any kind of VoIP failure that makes it impossible for a staff member to use their VoIP phone, they need to be trained to take the following actions in case emergency help is needed:

1. Use a cell phone or neighbor's desk phone (if working).

2. Use a non-VoIP emergency/courtesy phone located at designated points on each floor.
3. Pull the fire alarm.
4. For non-emergency situations, use email or instant messaging to communicate

For a major outage of VoIP or Rolm PBX service, the Emergency Management Operations Team (EMOT) will be activated.

20.2 Physical and Logical Network Infrastructure Failures

Studebaker building becomes unusable

In the event the Studebaker building becomes unusable for an extended period of time, staff will need to relocated as part of a larger business continuity plan that is not within the scope of the VoIP project.

In those relocated areas, IP phones can be deployed from our repair inventory for a small percentage of the 700 phones while additional phones are shipped from our suppliers. Using the web Provisioning Tool, users or Designers can individually forward VoIP extensions to cell phones or other alternate numbers (home, Rolm PBX, etc.) or the Network Infrastructure Software Systems group can make bulk changes in the provisioning database.

Failure of Studebaker building network infrastructure

In the event of a long-term building network infrastructure failure, use of the IP network will be unavailable for other purposes than just phones and it will likely render the building unusable. This type of failure should be considered equivalent to the case of the building becoming unusable, above.

Failure of both redundant outside plant fiber links

If the outage is caused by simultaneous interruption of both the 12th Avenue and Broadway (RCN) fiber loops by physical damage to the exterior under-street conduits, repair of the conduits and pulling and splicing replacement fiber will be performed by a contractor under retainer for 7x24 response. CUIT is in the process of establishing this contract and we expect to have it in place by June 2007.

Internal fiber link failures (e.g. within the Studebaker building or others downstream that the fiber routes through) will be repaired by CUIT Network Field Services staff or contractors as appropriate. Spare fiber cable, necessary tools (including a fusion splicer and time-domain reflectometer), accessories, and training are in place.

If a single fiber path failure occurs and repair is likely to take longer than several weeks before redundancy can be restored (e.g. due to a major disruptive event that destroys a large section of the conduit path), CUIT Network Infrastructure will deploy a backup microwave link system at a cost of approximately $50,000. This will take several weeks and will require an emergency Purchase Order.

Low Library building failure

If Low Library is significantly damaged, the entire Rolm PBX user community will be without phone service. VoIP users will lose direct inward dialing since those numbers currently route via the Rolm. Network Infrastructure Op-

erations will work with our TDM and ITSP providers to swing DID numbers over to replacement trunks that route via IP to the VoIP proxies. This work will also be used to restore service to current non-VoIP critical numbers at the University by replacing those phones with VoIP phones and/or off-system forwarding to cell phones or Verizon lines.

Simultaneous Failure of 103 Philosophy Hall and Computer Center (IP network)

If both data centers fail simultaneously or some logical error causes the network routers to fail (e.g. a zero-day attack on the routers), critical VoIP proxies will be unavailable, as will the University network which is dual-homed into the same two locations. This is an issue that is larger than the VoIP system. The Rolm PBX can be used to take over unavailable VoIP extensions since DID numbers are currently routed via the PBX.

A further mitigating option is available which involves locating a third redundant VoIP proxy at an off-site location such as 32 Avenue of the Americas or in Syracuse. The requirement for this additional step of risk mitigation needs to be further discussed in a larger business continuity context.

Internet Failure

If our redundant connectivity to the Internet completely fails, SIP trunking for in and outbound calls will not be available. TDM trunks will be used automatically although there will be some seconds of timeout delay while alternate routes are selected. As we transition our DID service from legacy TDM fully to SIP trunks, this risk will become greater and we will mitigate it by retaining some TDM trunks.

DNS, DHCP

DNS and DHCP failures will have a similar impact to a total campus network outage, however, the symptoms may appear to be sporadic or have a delayed onset. Polycom phones get their IP addresses from DHCP servers with long (8 hour) leases. DNS is used by the phone to perform NAPTR lookups to find the IP addresses of the proxies. Proxies are configured with IP addresses rather than DNS names to limit their dependence on DNS.

20.3 VoIP Application Failures

All proxies down

In the event that all VoIP proxies fail due to a common Linux OS or proxy software bug or exploit, the VoIP service will be rendered unusable. General mitigating actions will need to be taken while the Rolm PBX is used to define critical extensions that will be rehomed to the PBX. By the time the PBX is retired we will have amassed sufficient operational experience to determine the likelihood of this kind of failure and a mitigation strategy.

Voicemail application down

Failure of the voicemail application, while an inconvenience and potential issue for business continuity, is not considered a critical failure. If the application is unavailable, callers will receive an error tone.

Asterisk servers down

If the redundant Asterisk servers fail, media services such as voicemail, error recordings ("You have reached a non-working number..."), multiparty conferencing, and similar applications will be unavailable and replaced by error tone until service is restored.

Provisioning tool or file service down

If the Provisioning Tool fails, CUIT Design staff will be unable to implement moves, adds, and changes. However, the NOC and Network Software Systems staff will have the ability to manually make configuration changes to the OpenSER, Asterisk and Polycom phone configurations.

The provisioning file service (HTTP) is used by phones to check for new or updated configuration files at boot time and nightly. If the phone fails to reach the server or to find its own *MAC*.cfg file on the server, it will continue to use the local operational configuration installed in its flash memory. In the event of an emergency need for a change to a phone configuration, Design, NOC or Network Software and Systems staff can instruct the customer on how to manually configure the phone through its built-in configuration menu system.

20.4 Rolm PBX Service Failures

PSTN trunks down

If in- and outbound carrier PSTN trunks fail to the Rolm PBX, VoIP (and all Rolm) users will lose inbound and outbound service. Rerouting via existing ITSP providers can be used to provide outbound service. DID rerouting is largely dependent on the carrier (Verizon) and we are working to achieve some better disaster resiliency with an alternate carrier (PAETEC) using SIP trunking.

PBX failure

In the event of a system failure of all 9 Rolm 9751 nodes or the HiCom 300, actions described above for PSTN trunk failure will be performed. In addition, key Rolm extensions will be implemented as VoIP extensions.

Media Gateway failure

If both redundant media gateways between the HiCom 300 and VoIP service fail, routing of calls between VoIP and Rolm PBX users as well as DID calls for VoIP users will be disrupted. Using SIP trunking via our ITSPs and VoIP LCR routing, calls to 851, 853 and 854 numbers (but not 7-xxxx internal extensions) will be automatically rerouted as "long distance" calls via our ITSPs using the VoIP least cost routing capability, as long as all DIDs are routed to the Rolm first. The reverse will not be possible due to this routing.

Rolm PhoneMail down

Failure of the PhoneMail application, while an inconvenience and potential issue for business continuity, is not considered a critical failure. If the application is unavailable, calls will go unanswered.

21 References

References

[1] J. Rosenberg, Henning Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol. RFC 3261, Internet Engineering Task Force, June 2002.

[2] Administrator's Guide: SoundPoint/SoundStation IP SIP Version 2.0, August 2006.

[3] Asterisk: An Open Source PBX and telephony toolkit. http://www.asterisk.org/ (accessed 04/24/07).

[4] Jonathan Lennox, Xiaotao Wu, and Henning Schulzrinne. Call Processing Language (CPL): A Language for User Control of Internet Telephony Services. RFC 3880, Internet Engineering Task Force, October 2004.

[5] Steph Tryphonas, Daniel C. Burnett, Peter Danielsen, Bruce Lucas, Jim Ferrans, Jerry Carter, Scott McGlashan, Ken Rehor, Brad Porter, and Andrew Hunt. Voice Extensible Markup Language (VoiceXML) Version 2.0. Recommendation voicexml20-20040316, World Wide Web Consortium, March 16 2004.

[6] Cfengine: A Configuration Engine. http://www.cfengine.org (accessed 04/08/07).

[7] Revision Control Systems (RCS). http://www.gnu.org/software/rcs/ (accessed 04/08/07).

[8] Sylantro Systems Corp. SIP for Business: Delivering Business Class Features to SIP Phones. http://www.sylantro.com/solutions_sip.html (accessed 11/26/2006).

[9] Sylantro Systems Corp. SIP Implementation & Call Flows for Business Telephony Features. http://www.sylantro.com/solutions_sip.html (accessed 03/20/2007), May 2006.

[10] R. Mahy, B. Biggs, and R. Dean. The Session Initiation Protocol (SIP) Replaces Header. RFC 3891, Internet Engineering Task Force, September 2004.

[11] R. Sparks. The Session Initiation Protocol (SIP) Refer Method. RFC 3515, Internet Engineering Task Force, April 2003.

[12] Anurag Kumar and V. Venkataramanan. Implementing Bridged Line Appearances Using Session Initiation Protocol (SIP). Internet draft, Internet Engineering Task Force, June 2003. Work in progress.

[13] Adam Roach. Session Initiation Protocol (SIP)-Specific Event Notification. RFC 3265, Internet Engineering Task Force, June 2002.

[14] Jonathan Rosenberg, Henning Schulzrinne, and Rohan Mahy. An INVITE-Initiated Dialog Event Package for the Session Initiation Protocol (SIP). RFC 4235, Internet Engineering Task Force, November 2005.

[15] C. Jennings, J. Peterson, and M. Watson. Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks. RFC 3325, Internet Engineering Task Force, November 2002.

[16] R. Sparks. The Session Initiation Protocol (SIP) Referred-By Mechanism. RFC 3892, Internet Engineering Task Force, September 2004.

[17] Alan Johnston. Requirements and Implementation Options for the Multiple Line Appearance Feature using the Session Initiation Protocol (SIP). Technical report, February 27 2007.

[18] W. Marshall et al. SIP Extensions for Network-Asserted Caller Identity and Privacy within Trusted Networks. Internet draft, Internet Engineering Task Force, March 2002. Work in progress.

[19] J. Rosenberg and Henning Schulzrinne. An Offer/Answer Model with Session Description Protocol (SDP). RFC 3264, Internet Engineering Task Force, June 2002.

[20] A. Johnston and R. Sparks. Session Initiation Protocol Service Examples. Internet Draft draft-ietf-sipping-service-examples-05, Internet Engineering Task Force, September 2003. Work in progress.

[21] B. Campbell and R. Sparks. Control of Service Context using SIP Request-URI. RFC 3087, Internet Engineering Task Force, April 2001.

[22] R. Mahy. A Message Summary and Message Waiting Indication Event Package for the Session Initiation Protocol (SIP). RFC 3842, Internet Engineering Task Force, August 2004.

[23] Henning Schulzrinne and S. Petrack. RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals. RFC 2833, Internet Engineering Task Force, May 2000.

[24] M. Baugher, D. McGrew, M. Naslund, E. Carrara, and K. Norrman. The Secure Real-time Transport Protocol (SRTP). RFC 3711, Internet Engineering Task Force, March 2004.

[25] National Emergency Number Association. NENA Standard Formats & Protocols for ALI Data Exchange, ALI Response & GIS Mapping. http://www.nena9-1-1.org/9-1-1TechStandards/Standards_PDF/NENA%2002-010%20Standard%20A (accessed 11/27/2006).

[26] Henning Schulzrinne. Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information. RFC 4776, Internet Engineering Task Force, November 2006.

[27] Paul Congdon and David Frattura. LLDP-MED simplifies VoIP deployments. http://www.networkworld.com/news/tech/2004/110104techupdate.html (accessed 12/10/2006), Nov 2004. Network World.

[28] Texas A&M University. NG911 Project. http://ng911.tamu.edu (accessed 11/27/2006).

[29] Henning Schulzrinne. Emergency Services for Internet Telephony based on the Session Initiation Protocol (SIP). Internet draft, Internet Engineering Task Force, January 2003. Work in progress.

[30] ISDN Network Layer Protocol for Signaling. Recommendation Q.931, ITU, May 1998.

[31] P. Faltstrom and M. Mealling. The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM). RFC 3761, Internet Engineering Task Force, April 2004.

[32] J. Peterson, H. Liu, J. Yu, and B. Campbell. Using E.164 numbers with the Session Initiation Protocol (SIP). RFC 3824, Internet Engineering Task Force, June 2004.

[33] S. Levy, B. Byerly, and Jiong Yang. Diversion Indication in SIP. Internet draft, Internet Engineering Task Force, January 2003. Work in progress.

[34] Cisco Systems Inc. SIP Diversion Header Implementation for Redirecting Number. http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121rel/sipcfs/hennigan. (accessed 12/2/2006).

[35] wikipedia. VoIP spam. http://en.wikipedia.org/wiki/VoIP%5fspam (accessed 12/8/2006).

[36] S. Niccolina, S. Tartarelli, M. Stiemerling, and S. Srivastava. SIP Extensions for SPIT identification. Internet Draft draft-niccolini-sipping-feedback-spit-02, Internet Engineering Task Force, August 2006.

[37] Souhwan Jung, Jaeduck Choi, Youjae Won, and Young Duk Cho. Authentication between the Inbound Proxy and the UAS for Protecting SPIT in the Session Initiation Protocol (SIP). Internet Draft draft-jung-sipping-authentication-spit-00, Internet Engineering Task Force, October 2006.

[38] FCC. Communications Assistance for Law Enforcement Act. http://www.fcc.gov/calea (accessed 12/8/2006).

[39] Lawfully Authorized Electronic Surveillance. Technical Report J-STD-025 Rev. A, Telecommunications Industry Association, May 2000.

[40] Federal Communications Commissions. Communications Assistance for Law Enforcement Act and Broadband access and Services: Second Report and Order and Memorandum Opinion and Order. http://www.acuta.org/?1489 (members only; accessed 12/5/2006); also available via http://www.fcc.gov, May 2006.

[41] American Council on Education. The Application of CALEA to Higher Education Networks. http://www.educause.edu/ir/library/pdf/EPO0654.pdf (accessed 12/5/2006).

[42] Doug Carlson. Thinking Through the CALEA Exempt/Non-Exempt Issue. http://www.educause.edu/ir/library/pdf/CSD4607.pdf (accessed 12/5/2006).

[43] EDUCAUSE Networking Policy Council. Letter from the Networking Policy Council, August 2006.

[44] ACUTA. ACUTA Alert: Regulatory Environment for VoIP Providers. http://www.acuta.org/?1538 (members only; accessed 12/5/2006), August 2006.

[45] Summary of FCC CALEA Order by Wiley Rein and Fielding LLP. http://www.acuta.org/?1493 (members only; accessed 12/5/2006), May 2006.

[46] Henning Schulzrinne, Stephen Casner, Ron Frederick, and Van Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 3550, Internet Engineering Task Force, July 2003.

[47] Timur Friedman, Ramon Caceres, Alan Clark, and Eds. RTP Control Protocol Extended Reports (RTCP XR). RFC 3611, Internet Engineering Task Force, November 2003.